

Ein Beitrag für den methodischen Test von automatisierten Fahrfunktionen  
mit Hilfe von virtuellen Umgebungen

Von der Fakultät für Elektrotechnik, Informationstechnik, Physik der  
Technischen Universität Carolo-Wilhelmina zu Braunschweig

zur Erlangung des Grades eines Doktors

der Ingenieurwissenschaften (Dr.-Ing.)

genehmigte Dissertation

von Fabian Schuldt

aus Gifhorn

eingereicht am: 30.10.2016

mündliche Prüfung am: 13.02.2017

1. Referent: Professor Dr.-Ing. Markus Maurer
2. Referent: Professor Dr. rer. nat. Hermann Winner

Druckjahr 2017



# Danksagung

Die vorliegende Arbeit entstand während meiner Zeit als wissenschaftlicher Mitarbeiter am Institut für Regelungstechnik der Technischen Universität Braunschweig. In dieser Zeit hatte ich die Möglichkeit, viele interessante Menschen kennenzulernen.

Im Jahre 2011, zu Beginn des Forschungsprojekts *UR:BAN*, verstärkte ich das Team im Forschungsprojekt des Engstellenassistenten in der Volkswagen Konzernforschung. Für die interessante Zeit im Projekt möchte ich mich bei meinen Kollegen Sven Chlosta und Jochen Marx bedanken. Bedanken möchte ich mich auch besonders bei meinem Kollegen und nun auch Freund Sönke Freter für die vielen Dinge, die ich lernen durfte, und die vielen lustigen Stunden im *BArt*. Ein besonderer Dank geht an Dr. Stephan Scholz, der neben der Projektleitung auch die industrielle Betreuung meiner Dissertation übernommen hat. Weiterhin danke ich Dr. Lutz Junge und Dr. Christian Jördens für ihre Unterstützung.

Vielen Dank auch an die Kolleginnen und Kollegen Michaela Steinhoff, Kathrin Symkenberg und Marco Langerwisch aus der sogenannten VTD-Ecke. Mein Dank geht auch an Jens Krause für die Unterstützung zur Durchführung meiner Arbeit. Besonders bedanken möchte ich mich bei Marius Dupius für die zahlreich gelösten Tickets und den stets schnellen Support. Ohne dies wäre diese Arbeit nicht möglich gewesen.

Für die Diskussionen, Anregungen, Kritiken und Unterstützungen in den letzten vier Jahren am Institut für Regelungstechnik bedanke ich mich herzlichst bei meinen ehemaligen Kolleginnen und Kollegen Andreas Reschka, Sven Böhme, Jens Rieken, Simon Ulbrich, Peter Bergmiller, Richard Matthaei, Till Menzel, Marcus Nolte, Gerrit Bagschik, Torben Stolte, Stefanie Scheffer, Veronika Krapf, Johanna Rieke, Martin Escher, Jan Timo Wendler, Horea Cernat und Susanne Ernst. Es war mir eine Ehre mit euch zusammenzuarbeiten. Mein Dank geht auch an die von mir betreuten Studierenden, die eine wissenschaftliche Arbeit bei mir geschrieben haben und somit einen entscheidenden Teil zu dieser Arbeit beigetragen haben.

Prof. Dr.-Ing. Markus Maurer und Prof. Dr.-Ing. Bernd Lichte danke ich für die wissenschaftliche Betreuung und die zahlreichen Diskussionen während meiner Promotionszeit. Durch eure Ratschläge und Kritiken hat sich meine wissenschaftliche Laufbahn stets in die richtige Richtung entwickelt. Prof. Dr. rer. nat. Hermann Winner danke ich für die Übernahme des Korreferats.

Ein besonderer Dank geht an meine Familie und meine Freunde für ihr Verständnis und ihre Unterstützung in jeglicher Hinsicht. Besonders danken möchte ich meiner Frau Sandra, die mich immer unterstützt hat und für jede Überstunde für die Dissertation Verständnis gezeigt hat.

Seershausen, im November 2016  
*Fabian Schuldt*





# Kurzfassung

In den vergangenen Jahren haben zahlreiche Projekte und Demonstrationen die öffentliche Aufmerksamkeit auf das automatisierte Fahren gelenkt. Der Fokus lag dabei vor allem auf der funktionalen Entwicklung der Fahrfunktionen. Die Demonstrationen lassen auf eine nahe Serieneinführung der Systeme schließen. Jedoch wurde durch die Demonstrationen lediglich die funktionale Sicht auf die Fahrfunktion gezeigt. Das Testkonzept, welches für die Freigabe erforderlich ist, stand nicht im Fokus.

In der vorliegenden Arbeit wird ein Beitrag für den methodischen Test von automatisierten Fahrfunktionen mit Hilfe von virtuellen Umgebungen geleistet. Im ersten Teil wird die Notwendigkeit eines systematischen Testkonzepts begründet. Dazu wird das Testdilemma automatisierter Fahrfunktionen dargestellt und die These aufgestellt, dass ein szenariobasiertes Testkonzept eine mögliche Lösung für das Testdilemma darstellen könnte. Es werden sechs Forschungsfragen aufgestellt, die für die Entwicklung eines szenariobasierten Ansatzes beantwortet werden müssen.

Im zweiten Teil werden die Grundlagen und Voraussetzungen der vorliegenden Arbeit dargestellt. Es werden Begriffe und Definitionen für den Bereich des Tests, zur Beschreibung von Schnittstellen und zur Beschreibung von Straßen vorgestellt. Weiterhin wird der Begriff der Komplexität von Szenarien untersucht. Es werden eine Beschreibung von Automatisierungsgraden und eine funktionale Systemarchitektur für automatisierte Fahrfunktionen vorgestellt. Der Teil schließt mit Methoden des Softwaretests sowie der Verkehrssimulation und einer Klassifikation von verschiedenen X-in-the-Loop-Verfahren ab.

Im dritten Teil wird das Testkonzept des modularen virtuellen Testbaukastens vorgestellt. Es werden Anforderungen definiert sowie der Aufbau und die Schnittstellen zwischen den Modulen des Testbaukastens präsentiert. Für die Auswahl und Analyse der Einflussparameter, die Testfallerstellung und die Testdurchführung mittels X-in-the-Loop-Verfahren werden Anforderungen definiert und der relevante Stand der Technik vorgestellt. Aus dem vorgestellten Stand der Technik wird der Forschungsbedarf abgeleitet. Für die Auswahl und Analyse der Einflussparameter wird ein Schema zur Beschreibung der Einflussparameter hergeleitet und es werden Informationsquellen für die Auswahl und Analyse von Einflussparametern bewertet. Für die Testfallerstellung wird ein generisches Modell zur Beschreibung von Szenarien vorgestellt. Weiterhin wird eine kombinatorische Testfallableitung präsentiert. Für die Testdurchführung wird eine Zuordnungsmethode für Testfälle auf verschiedene X-in-the-Loop-Verfahren beschrieben. Zusätzlich werden Testtreiber für die Module einer funktionalen Systemarchitektur analysiert und die Testtreiber des modularen virtuellen Testbaukastens vorgestellt. Für das Modul der Testfallauswertung werden Anforderungen definiert und Methoden aus dem Stand der Technik zur Bewertung und zur Analyse der Testergebnissen präsentiert. Der Teil schließt mit einer Beschreibung der Limitationen des Testbaukastens ab.

Der vierte Teil beschreibt die Anwendung des Testbaukastens im Fallbeispiel des Engstellenassistenten. Das Projekt wird vorgestellt und die verschiedenen Module des Testbaukastens werden angewendet.



# Abstract

In the past years, many projects and demonstrations have highlighted the public's interest in automated driving functions. These projects were mostly focused on functional development and demonstrate a possible series production of the automated driving functions. However, the demonstrations show only the functional view of automated driving and didn't concentrate on test process for such systems.

This dissertation contributes to the systematical test of driving functions with virtual environments. The first part establishes the necessity of a systematic test concept for automated driving functions. The challenge of testing automated driving functions is presented and the assumption that scenario-based test concept can be a possible solution. Six research questions will be asked in this section, which have to be answered for the development of a scenario-based test concept.

The second part defines important terms and analyses prior art as a foundation for this dissertation. Furthermore the levels of automated driving functions are presented and a functional system architecture is suggested. Finally, methods of software testing, traffic simulations, and classification methods for x-in-the-loop techniques are discussed.

The third part purposes a concept for a modular virtual testing toolbox. The structure and interfaces between modules of the toolbox are described. Furthermore, requirements are stated for the following modules: selection and analysis of influence parameters, test case generation, test execution with x-in-the-loop techniques, and test case evaluation. For each of these modules selected state of the art methods are presented. Hence, the need for research is deduced. For the selection and analysis of influence parameters, a schema for describing influence parameters is introduced. Furthermore, resources for the selection and analysis of influence parameters are evaluated. For the test case generation, a unified model for the description of scenarios is presented. Additionally, a combinatorial test case deduction is described. For the test case execution, a method for assigning test cases to x-in-the-loop techniques is suggested. For the test case evaluation, a method for the evaluation and analysis of the test result is presented. A review of the limitation of the modular virtual testing toolbox closes this part.

The fourth part presents the application of the modular virtual test toolbox to the constriction assistance system. The project is described and the single modules of the toolbox are applied to the assistance system.



# Inhaltsverzeichnis

	Seite
<b>I Einführung</b>	<b>1</b>
<b>1 Einleitung</b>	<b>3</b>
1.1 Motivation . . . . .	3
1.2 Testdilemma automatisierter Fahrfunktionen . . . . .	4
1.3 Identifikation des Forschungsbedarfs für ein szenariobasiertes Testkonzept .	7
1.4 Aufbau der vorliegenden Arbeit . . . . .	12
<b>II Grundlagen und Voraussetzungen</b>	<b>15</b>
<b>2 Terminologie der vorliegenden Arbeit</b>	<b>17</b>
2.1 Allgemeine Terminologie . . . . .	17
2.2 Begriffe des Testens . . . . .	22
2.3 Begriffe zur Beschreibung von Schnittstellen . . . . .	24
2.4 Begriffe des Straßenaufbaus . . . . .	36
<b>3 Automatisierte Fahrfunktionen</b>	<b>41</b>
3.1 Funktionale Beschreibung von Automatisierungsgraden . . . . .	41
3.2 Funktionale Systemarchitektur für automatisierte Fahrfunktionen . . . .	46
3.3 Exemplarische Fahrfunktion zur Veranschaulichung des Testkonzepts . . .	51
<b>4 Methoden des Softwaretests und der Verkehrssimulation</b>	<b>53</b>
4.1 Softwaretestmethoden . . . . .	53
4.2 Verkehrssimulationen . . . . .	58
4.3 Klassifikation von X-in-the-Loop-Verfahren . . . . .	60
4.4 Zusammenfassung . . . . .	70
<b>III Der modulare virtuelle Testbaukasten</b>	<b>71</b>
<b>5 Anforderungen an den modularen virtuellen Testbaukasten</b>	<b>73</b>
5.1 Anforderungen an ein Testkonzept für automatisierte Fahrfunktionen . . .	73

5.2	Anforderungen durch Anwender . . . . .	77
5.3	Anforderungen durch Testobjekte . . . . .	77
<b>6</b>	<b>Aufbau und Schnittstellen des modularen virtuellen Testbaukastens</b>	<b>81</b>
6.1	Aufbau des Testbaukastens . . . . .	81
6.2	Schnittstellen zwischen den Modulen . . . . .	83
<b>7</b>	<b>Auswahl und Analyse von Einflussparametern</b>	<b>85</b>
7.1	Anforderungen . . . . .	85
7.2	Stand der Technik . . . . .	87
7.3	Informationsquellen zur Auswahl von Einflussparametern . . . . .	90
7.4	Informationsquellen zur Analyse von Einflussparametern . . . . .	94
7.5	Schema zur Darstellung von Einflussparametern . . . . .	96
7.6	Zusammenfassung . . . . .	100
<b>8</b>	<b>Testfallerstellung</b>	<b>103</b>
8.1	Anforderungen . . . . .	103
8.2	Stand der Technik . . . . .	104
8.3	Ein generisches 4-Ebenen-Modell für die Beschreibung von Szenarien . . .	109
8.4	Systematische Testfallableitung . . . . .	119
8.5	Zusammenfassung . . . . .	125
<b>9</b>	<b>Testdurchführung mittels X-in-the-Loop-Verfahren</b>	<b>127</b>
9.1	Anforderungen . . . . .	127
9.2	Stand der Technik . . . . .	132
9.3	Eine Methode für die Zuordnung von Testfällen auf X-in-the-Loop-Verfahren	140
9.4	Testtreiber für die Module der funktionalen Systemarchitektur . . . . .	144
9.5	Zusammenfassung . . . . .	157
<b>10</b>	<b>Testfallauswertung</b>	<b>159</b>
10.1	Anforderungen . . . . .	159
10.2	Methoden aus dem Stand der Technik . . . . .	160
10.3	Zusammenfassung . . . . .	165
<b>11</b>	<b>Limitationen des modularen virtuellen Testbaukastens</b>	<b>167</b>
11.1	Erfüllung der gestellten Anforderungen an ein Testkonzept . . . . .	167
11.2	Erfüllung der gestellten Anforderungen durch Anwender . . . . .	169
11.3	Erfüllung der gestellten Anforderungen durch Testobjekte . . . . .	170
11.4	Limitationen der Module des Testbaukastens . . . . .	171

<b>IV</b>	<b>Die Anwendung des modularen virtuellen Test- baukastens</b>	<b>175</b>
<b>12</b>	<b>Der Engstellenassistent</b>	<b>177</b>
12.1	Projektbeschreibung . . . . .	177
12.2	Auswahl und Analyse von Einflussparametern . . . . .	181
12.3	Testfallerstellung . . . . .	184
12.4	Testdurchführung . . . . .	189
12.5	Testfallauswertung . . . . .	194
12.6	Zusammenfassung . . . . .	200
<b>V</b>	<b>Zusammenfassung und Ausblick</b>	<b>203</b>
<b>13</b>	<b>Zusammenfassung und Ausblick</b>	<b>205</b>
13.1	Zusammenfassung . . . . .	205
13.2	Identifizierter Forschungsbedarf . . . . .	206
13.3	Ausblick . . . . .	210
	<b>Eigene Veröffentlichungen</b>	<b>211</b>
	<b>Betreute studentische Arbeiten</b>	<b>213</b>
	<b>Literaturverzeichnis</b>	<b>215</b>
	<b>Anhang</b>	<b>233</b>
<b>A</b>	<b>Detaillierte Ergebnisse der Fallstudie des Engstellenassistenten</b>	<b>235</b>





# Teil I

## Einführung



# 1 Einleitung

## 1.1 Motivation

Fahrerassistenzsysteme mit maschineller Wahrnehmung, wie sie von Stiller (2005) und Maurer (2015a) vorgestellt werden, haben in den letzten Jahren Serienreife erlangt und sind mittlerweile fester Bestandteil von vielen Fahrzeugklassen (Winner, 2015a, S.69, S.482, S.592, S.889). Im Gegensatz zu konventionellen Fahrerassistenzsystemen nutzen Fahrerassistenzsysteme mit maschineller Wahrnehmung Sensoren, um die Umgebung zu erfassen. Als Beispiel ist der Abstandsregeltempomat (engl. Adaptive Cruise Control) (ISO 15622, 2010; ISO 22179, 2009) zu nennen, welcher die Zeitlücke zum vorausfahrenden Fahrzeug regelt. Ein Überblick über die Entwicklung des Systems wird von Winner (2003) gegeben. Aktuelle Systemausprägungen werden von Winner und Schopper (2015) vorgestellt. Ein weiteres, in der Serie etabliertes System ist der Fahrstreifenhalteassistent (engl. Lane Keeping Assistance System) (ISO 11270, 2014), der den Fahrer bei der Querführung unterstützt. Unter anderem wird dieses Querführungssystem von Bartels u. a. (2015) vorgestellt. Aktuelle Funktionsausprägungen des Systems, wie im 7er BMW (BMW, 2016) oder in der E-Klasse von Mercedes (Kacher, 2016), zeigen, dass das System bei verfügbaren Fahrstreifenmarkierungen oder vorausfahrenden Fahrzeugen in den Verkehrsdomänen Autobahn, Landstraße und Stadtstraße funktioniert. Insgesamt ist zu beobachten, dass die entwickelten Fahrfunktionen in steigendem Umfang in die Längs- und/oder Querführung des Fahrzeugs eingreifen.

Bei den genannten Systemen handelt es sich um Assistenzsysteme, bei denen nach SAE (2014)<sup>1</sup> entweder die Längs- oder die Querführung von der Fahrfunktion übernommen wird. Fahrfunktionen mit einem höheren Automatisierungsgrad, bei denen die Längs- und Querführung des Fahrzeugs dauerhaft von der Fahrfunktion übernommen werden, befinden sich nur zum Teil in Serie. Ein Beispiel ist die von Tesla genannte *Autopilot*-Funktion<sup>2</sup>, welche dauerhaft die Längs- und Querführung des Fahrzeugs übernimmt. Der Fahrer muss jedoch dauerhaft das System überwachen. Systeme der Automatisierungsgrade 3 bis 5 nach Einstufungen der SAE (2014) befinden sich derzeit nicht in Serie. Dies liegt unter anderem an fehlenden Testmethoden, die für eine Freigabe solcher Systeme benötigt werden. Maurer (2013) beschreibt die Herausforderung der Freigabe der zukünftigen automatisierten Fahrfunktionen mit den Worten:

„Wir können aufhören weitere Fahrfunktionen zu entwickeln, wenn wir keine Methoden finden, um die Systeme zu testen und abzusichern.“

Aktuell startende Forschungsprojekte, wie das Projekt *PEGASUS*<sup>3</sup> (Plättner u. a., 2016), welches Methoden zur Absicherung von automatisierten Fahrfunktionen als Forschungsschwerpunkt adressiert, zeigen ebenfalls die Relevanz der Forschung nach neuen Testmethoden.

---

<sup>1</sup>Auf die verschiedenen Automatisierungsgrade nach SAE (2014) wird in Kapitel 3 eingegangen.

<sup>2</sup><http://www.sueddeutsche.de/auto/autopilot-im-tesla-model-s-einfach-mal-loslassen-1.2723051>, abgerufen: 16.04.2016

<sup>3</sup>PEGASUS: Projekt zur Etablierung von generell akzeptierten Gütekriterien, Werkzeugen und Methoden sowie Szenarien und Situationen zur Freigabe hochautomatisierter Fahrfunktionen

Bisher lag der Schwerpunkt der Forschung darin, die technische Realisierbarkeit der automatisierten Fahrfunktionen zu demonstrieren. Dies zeigen unter anderem die Demonstrationen von Audi<sup>4</sup> und Daimler (Ziegler u. a., 2014). Die Erforschung von Methoden zur Absicherung der Fahrfunktionen stand bisher nicht im Mittelpunkt. Dies sollte sich allerdings ändern, da in Zukunft neben der technischen Realisierbarkeit einer automatisierten Fahrfunktion auch die Testbarkeit gezeigt werden sollte, damit die Fahrfunktionen von den Entwicklungsabteilungen in Serie gebracht werden können. Somit sollten die Forschungsabteilungen der Automobilhersteller diesen Trend für eine Weiterentwicklung ihres Forschungsschwerpunktes nutzen. Insgesamt steigt die Nachfrage nach neuen Testmethoden für automatisierte Fahrfunktionen, wodurch dieses Thema in den Fokus der Forschungsabteilungen der Automobilhersteller rücken müsste. Diese Nachfrage nach neuen Testmethoden ist die Motivation der vorliegenden Arbeit, die ein Testkonzept für automatisierte Fahrfunktionen vorstellt. Im folgenden Kapitel werden dazu das Testdilemma der automatisierten Fahrfunktionen beschrieben und die sich daraus ergebenden Forschungsfragen erläutert.

## 1.2 Testdilemma automatisierter Fahrfunktionen

Für einen sicheren Einsatz automatisierter Fahrfunktionen im öffentlichen Straßenverkehr sollte vor der Serieneinführung ein technisch und funktionaler erforderlicher Reifegrad der Fahrfunktion nachgewiesen werden, wie er beispielsweise für elektronische Systeme von der ISO 26262 (2011) gefordert wird. Diese Nachweise wurden für Assistenzfunktionen durch den Test im öffentlichen Straßenverkehr und auf Testgeländen erbracht. Dieses Vorgehen wurde anhand der Freigaben der Systeme des Abstandsregeltempomaten (ISO 15622, 2010; ISO 22179, 2009) und des Fahrstreifenhalteassistenten (ISO 11270, 2014) demonstriert. Die Systeme werden dabei über eine fehlerfreie Funktionalität innerhalb einer definierten Kilometerlaufleistung abgesichert. Weiterhin kann der *Code of Practice for the Development of ADAS* (Übersetzung: Leitfaden zur Entwicklung von Fahrerassistenzsystemen mit maschineller Wahrnehmung) (ERTICO-ITS Europe, 2009) für die Entwicklung der Systeme verwendet werden, um „eine angemessene Sicherheit zu erzielen und die geforderte Sorgfaltspflicht zu dokumentieren“ (Maurer, 2015b).

Der Nachweis der ausreichenden Sicherheit für eine Freigabe der Systeme kann bei automatisierten Fahrfunktionen jedoch nicht mehr ökonomisch erbracht werden. Dies skizziert die folgende Betrachtung von Wachenfeld und Winner (2015a). Ziel der Betrachtung ist die Beurteilung für den Testaufwand, mit dem gezeigt wird, dass ein automatisiert fahrendes Fahrzeug mit Automatisierungsgrad 5, nach SAE (2014), das Risiko im Straßenverkehr gegenüber einem menschlichen Fahrer<sup>5</sup> nicht erhöht. Ein Anwendungsfall für dieses Fahrzeug ist beispielsweise das von Wachenfeld u. a. (2015) vorgestellte *Fahrzeug auf Abruf*. Als Metrik wird die Zahl der Verkehrsunfälle in Deutschland herangezogen. Es werden als schlimmster Fall nur die Unfälle mit getöteten Verkehrsteilnehmern betrachtet. Nach den Zahlen des statistischen Bundesamts (Statistisches Bundesamt, 2016) ereigneten sich im Jahr 2014 3377 Unfälle mit Getöteten. Bei einer Fahrleistung von insgesamt 735 Milliarden Kilometern ergeben sich 217 Millionen Kilometer zwischen zwei Unfällen mit Todesfolge. Da dieser Wert nur eine Stichprobe darstellt und die Werte einer statistischen Verteilung unterliegen, muss für eine weitere Betrachtung eine statistische Verteilung gewählt werden. Von Wachenfeld

<sup>4</sup><http://blog.audi.de/2015/01/05/mit-jack-durch-kalifornien>, abgerufen: 11.04.2016

<sup>5</sup>Der Begriff des Fahrers wird in der vorliegenden Arbeit für menschliche Fahrerinnen und Fahrer verwendet.

und Winner (2015a) wird hierzu die Poisson-Verteilung gewählt, dargestellt mit Hilfe der Wahrscheinlichkeitsverteilung in Gleichung 1.1.

$$P_\lambda(k) = \frac{\lambda^k}{k!} e^{-\lambda} \quad (1.1)$$

$P_\lambda(k)$  beschreibt die Wahrscheinlichkeit für das Ereignis  $k$ , welches in diesem Fall die Anzahl der Unfälle mit getöteten Verkehrsteilnehmern darstellt.  $\lambda$  bezeichnet den Erwartungswert für das Ereignis, welcher durch den Quotienten in Gleichung 1.2 berechnet wird.

$$\lambda = \frac{s_{test}}{s_{leist}} \quad (1.2)$$

$s_{test}$  beschreibt die beobachteten Testkilometer und  $s_{leist}$  die Leistungsfähigkeit des Systems. Die Leistungsfähigkeit des Systems stellt die erwartete Anzahl der gefahrenen Kilometer zwischen zwei Unfällen mit Getöteten dar. Dabei wird angenommen, dass ein Fahrzeug mit automatisierter Fahrfunktion und ein Fahrzeug mit menschlichem Fahrer die gleiche Anzahl von Testkilometern  $s_{test} = a_s \cdot \bar{s}$  fahren.  $\bar{s}$  beschreibt die durchschnittliche Distanz zwischen zwei Unfällen mit getöteten Verkehrsteilnehmern. Der Faktor  $a_s$  wird zur Streckung der Strecke  $\bar{s}$  verwendet. Die Leistungsfähigkeit  $s_{leist} = a_{leist} \cdot \bar{s}$ , der in diesem Beispiel vollautomatisierten Fahrfunktion nach der Definition der SAE (2014), wird relativ zum Fahrzeug mit menschlichem Fahrer gewählt. Mit der Poisson-Verteilung wird berechnet, welche Teststrecke notwendig ist, um zu zeigen, dass eine automatisierte Fahrfunktion um einen gewissen Faktor in der Leistungsfähigkeit besser als ein menschlicher Fahrer ist. Von Wachenfeld und Winner (2015a, S.456) wird dazu beispielhaft ein Signifikanzniveau von 5% gewählt. Der Streckungsfaktor ist soweit zu erhöhen, bis die Wahrscheinlichkeit für das Ereignis unter dem Signifikanzniveau liegt, um die Leistungsfähigkeit nachzuweisen.

Bei der Annahme, dass die automatisierte Fahrfunktion hinsichtlich des Kriteriums *erwartete Anzahl gefahrener Kilometer zwischen zwei Unfällen mit Getöteten* doppelt so gut sein soll wie ein menschlicher Fahrer, ergibt sich nach Wachenfeld und Winner (2015a, S.457) ein Streckungsfaktor von  $a_s = 10$ . Damit ergibt sich ein Testaufwand von 2,17 Milliarden Testkilometer für die Absicherung einer Funktionsversion des automatisierten Fahrzeugs auf Abruf. Bei jeder Änderung der Funktionsversion (Softwarestand, Hardwareänderungen) müsste diese Distanz bei diesem streckenbasierten Testansatz für eine Absicherung wiederholt werden. Wenn sich das Fahrzeug mit der ausgewählten Fahrfunktion nur auf der Autobahn bewegt, ergibt sich ein Vergleichswert von 614 Millionen Kilometern zwischen zwei Unfällen und damit eine Testdistanz von 6,14 Milliarden Testkilometern. Für Systeme, die innerstädtische Szenarien und Szenarien auf der Landstraße fokussieren, würde nach der gewählten Abschätzung eine geringere Anzahl von Testkilometern berechnet werden, da in diesen Domänen eine geringere Distanz zwischen zwei Unfällen liegt<sup>6</sup>.

---

<sup>6</sup>Das statistische Bundesamt macht keine Angaben für die Fahrleistung auf städtischen Straßen. Daher kann nur eine gemeinsame Betrachtung von Stadtstraßen und Landstraßen vorgenommen werden. Die Anzahl der Getöteten auf innerstädtischen Straßen beträgt 983. Auf Landstraßen wurden 2019 Verkehrsteilnehmern getötet. Bei einer Fahrleistung von 500 Milliarden Kilometern auf innerstädtischen Straßen und Landstraßen ergibt sich ein Abstand von 168 Millionen Kilometern zwischen zwei Unfällen mit Getöteten (Statistisches Bundesamt, 2016). Dies ergibt somit eine Distanz von 1,68 Milliarden Testkilometern für diese Domänen.

Diese Abschätzung zeigt den Aufwand für einen Nachweis durch einen streckenbasierten Testansatz. Mit einem erfolgreichen Test kann gezeigt werden, dass die automatisierte Fahrfunktion eine definierte Leistungsfähigkeit gegenüber einem menschlichen Fahrer aufweist. Die Testtiefe beziehungsweise die Testabdeckung<sup>7</sup> wird mit diesem Ansatz jedoch nur ansatzweise geschätzt. Dies liegt an der Testart, die einem Zufallstest<sup>8</sup> ähnlich ist. Dadurch wird keine genaue Aussage über die Testabdeckung getroffen, da nicht bekannt ist, welche Szenarien während der Fahrt getestet werden. Somit kann, begründet durch die Testart, eine Erhöhung der Anzahl von Testkilometern nur bedingt eine Erhöhung der Testabdeckung erreichen. Weiterhin wird nicht gezeigt, dass während der Fahrt auf der Teststrecke die Anwendungsszenarien, die für eine Freigabe des Systems getestet werden müssten, auftreten beziehungsweise getestet werden. Daher wäre an dieser Stelle ein Testansatz, welcher auf Szenarien basiert, eine zu überprüfende Alternative zum streckenbasierten Ansatz. Ein szenariobasierter Ansatz würde gezielt die Anwendungsszenarien testen und variieren, um so eine systematische Erhöhung der Testabdeckung zu erreichen.

Die Anzahl der möglichen Anwendungsszenarien und damit Testfälle steigt mit der Komplexität der Umgebung. Die Komplexität ergibt sich hier aus der Anzahl der Elemente der Umgebung und deren Interaktionen untereinander. Für eine automatisierte Fahrfunktion, die für die Stadt bestimmt ist, würde dies bedeuten, dass für die Freigabe des Systems im Verhältnis zur Autobahn mehr Szenarien und mehr Variationen der Szenarien getestet werden müssten, da in der Stadt die Anzahl der Komponenten und deren Interaktion höher ist als auf einer Autobahn. Nach der Schätzung von Wachenfeld und Winner (2015a) müssten jedoch weniger Testkilometer gefahren werden, da in der Stadt der Abstand zwischen zwei Unfällen mit Getöteten geringer ist (Statistisches Bundesamt, 2016). Damit wäre ein streckenbasierter Ansatz in der Stadt *ergiebig* als auf der Autobahn. Umso mehr Unfälle in der Domäne auftreten, desto weniger Testkilometer werden für eine Freigabe benötigt. Jedoch kann eine hohe Anzahl von Unfällen auch ein Kennzeichen von hoher Komplexität der Umgebung sein. Dies zeigt beispielsweise der Vergleich der Unfallzahlen von Autobahnen und Stadtstraßen. Der Testaufwand sollte somit mit der Anzahl der möglichen Anwendungsszenarien skalieren.

Daher zeigt sich, dass andere Methoden für die Absicherung genutzt werden müssen. Da es aktuell keine anderen verfügbaren Methoden für die Absicherung von automatisierten Fahrfunktionen gibt, müssen neue Methoden entwickelt werden. Bengler u. a. (2014), welche Maurer und Winner (2013) zitieren, beschreiben die Herausforderungen der Freigabe der automatisierten Fahrfunktionen mit den Worten:

*„If testing and assessment methods cannot keep pace with this functional growth, they will become the bottleneck of the introduction of advanced DAS to the market.“*

Die oben dargestellte Diskrepanz zwischen dem strecken- und szenariobasierten Testansatz zeigt, dass ein szenariobasiertes Testkonzept eine zu überprüfende Alternative beziehungsweise Ergänzung zum streckenbasierten Testansatz darstellt. In einem szenariobasierten Ansatz werden im Gegensatz zum streckenbasierten Ansatz gezielt Szenarien für das Testobjekt abgeleitet und diese systematisch variiert. Dadurch wird mit Hilfe eines streckenbasierten Ansatz die Möglichkeit geschaffen, die Testabdeckung systematisch zu erhöhen. Virtuelle

---

<sup>7</sup>Der Begriff der Testabdeckung wird in Kapitel 2.2.1 vorgestellt

<sup>8</sup>Die Testart der Zufallstests werden unter anderem von Liggesmeyer (2009, S.208) vorgestellt.

Umgebungen können eingesetzt werden, um einen Teil der Tests in der Simulation durchzuführen und somit die Kosten für beide Ansätze zu reduzieren. Dies ist sowohl für den streckenbasierten als auch szenariobasierten Ansatz möglich. Die Vorteile der Simulation liegen jedoch in der Variationsmöglichkeit der Szenarioparameter. Dadurch kann ein szenariobasierter Ansatz größere Vorteil aus der Simulation ziehen, da für diesen die gezielte Variation der Szenarioparameter benötigt wird. Es wird daher die folgende These aufgestellt:

### **These**

*Ein szenariobasiertes Testkonzept, welches gezielt validierte virtuelle Umgebungen als Testwerkzeug einsetzt, bietet das Potential, die Herausforderung der Absicherung von zukünftigen automatisierten Fahrfunktionen durch eine systematische Erhöhung der Testabdeckung zu lösen.*

Der Forschungsbedarf, der sich aus der aufgestellten These ergibt, wird im folgenden Kapitel diskutiert und in Form von Forschungsfragen beschrieben. Allerdings sei bereits an dieser Stelle angemerkt, dass ein szenariobasiertes Testkonzept keine vollständige Testabdeckung erreichen kann. Da es sich bei automatisierten Fahrfunktionen um offene Systeme handelt, die sich in einer offenen Umgebung bewegen, kann nicht abgeschätzt werden, wie viele Szenarien insgesamt existieren. Dadurch kann derzeit keine vollständige Testabdeckung nachgewiesen werden. Die Messung der Testabdeckung stellt somit für den streckenbasierten als auch szenariobasierten Ansatz eine große Herausforderung dar und ist derzeit noch eine offene Forschungsfrage. Diese Forschungsfrage wird in dieser Arbeit nicht beantwortet. Jedoch wird untersucht, ob ein szenariobasiertes Testkonzept zum Beispiel gegenüber der oben beschriebenen zehnfachen Referenzstrecke eine systematische Erhöhung der Testabdeckung erreichen kann und somit einen Beitrag zur Forschung leistet. Weiterhin stellt die Validierung der virtuellen Umgebungen eine große Herausforderungen dar, die derzeit ebenfalls eine offene Forschungsfrage ist. Durch die Validierung der virtuellen Umgebung wird die Möglichkeit geschaffen, Realtests durch virtuell durchgeführte Tests zu ersetzen. Die Forschungsfrage der Validierung wird in dieser Arbeit nicht beantwortet.

## **1.3 Identifikation des Forschungsbedarfs für ein szenariobasiertes Testkonzept**

Im vorhergehenden Kapitel wurde das Testdilemma automatisierter Fahrfunktionen beschrieben und die These aufgestellt, dass ein szenariobasiertes Testkonzept das Potential bietet, eine Lösung für das Testdilemma darzustellen. In diesem Kapitel wird dazu der Forschungsbedarf in Form von Forschungsfragen beschrieben, um die These zu überprüfen. Insgesamt werden sechs Forschungsfragen aufgestellt, die das Ziel haben, die verschiedenen Herausforderungen der These zu beleuchten. Darauf folgend wird überprüft, ob die Arbeit von Schmidt (2012), die den Stand der Technik für ein Gesamttestkonzept für kamerabasierte Systeme darstellt, die gestellten Forschungsfragen beantwortet oder welcher Forschungsbedarf weiterhin besteht.



### **Forschungsfrage 1**

*Welchen Aufbau bzw. welche Bestandteile muss ein szenariobasiertes Testkonzept für automatisierte Fahrfunktionen haben?*

Die erste Frage beschäftigt sich mit dem Aufbau eines Testkonzepts für automatisierte Fahrfunktionen. Welche Bestandteile werden im Testkonzept benötigt, damit dieses systematisch, effizient und effektiv Testfälle ableitet, durchführt und auswertet? Weiterhin stellt sich die Frage, welche Schnittstellen zwischen diesen Bestandteilen geschaffen werden müssen, damit die Bestandteile Informationen austauschen können. Dies ist besonders wichtig, um die Austauschbarkeit von Methoden innerhalb der Bestandteile zu gewährleisten. Damit kann das Potential geschaffen werden, das Testkonzept an verschiedene Testobjekte anzupassen. Eine mögliche Lösung dieser Forschungsfrage wird in Kapitel 6 thematisiert.

### **Forschungsfrage 2**

*Wie können Einflussparameter für den Test von automatisierten Fahrfunktionen systematisch abgeleitet werden und welche Herausforderungen entstehen dabei?*

Die Identifikation von relevanten Einflussparametern für die Parametrisierung von Testfällen ist einer der entscheidenden Faktoren in einem Testkonzept für automatisierte Fahrfunktionen. Mit der Identifizierung der entscheidenden Einflussparameter besteht die Möglichkeit, die Repräsentativität der Szenarien und Testfälle für das Testobjekt herzustellen. Es stellt sich dabei die Frage, wann ein Parameter für das Testobjekt relevant ist und was bei einer Nichtbeachtung eines solchen Parameters geschieht. Weiterhin ist eine offene Fragestellung, welche Informationsquellen derzeit genutzt werden können, um zum einen die Einflussparameter zu identifizieren und zum anderen repräsentative Werte für die Parameter zu bestimmen. Dabei spielt die später diskutierte Diskretisierung von wertkontinuierlichen Parametern eine entscheidende Rolle für diese Forschungsfrage. Die Herausforderungen dieser Forschungsfrage und mögliche Lösungen werden in Kapitel 7 thematisiert.

### **Forschungsfrage 3**

*Wie kann ein Beschreibungsformat für Szenarien für automatisierte Fahrfunktionen aussehen und wie können aus diesen Szenarien Testfälle abgeleitet werden? Mit welchen Methoden kann die Anzahl der Testfälle bei einer bekannten Testabdeckung reduziert werden?*

Eine weitere Forschungsfrage beschäftigt sich mit dem Beschreibungsformat von Szenarien und der Ableitung von Testfällen aus diesen Szenarien. Das Beschreibungsformat soll soweit flexibel sein, dass die identifizierten Einflussparameter im Szenario beschrieben werden können. Dabei stellt sich die Frage, wie die Elemente des Szenarios gruppiert werden, um einen systematischen Aufbau zu bieten und eine systematische Variation der Szenarien zu ermöglichen. Weiterhin besteht die Forderung, dass die Möglichkeit besteht, die Szenarien so zu beschreiben, dass diese reproduzierbar durchführbar sind. Dies soll auch gelten, wenn das Verhalten des Testobjekts mit Unsicherheiten behaftet ist. Auf Basis der beschriebenen Szenarien sind konkrete Testfälle abzuleiten. Aufgrund des exponentiellen Wachstums der Anzahl von möglichen Kombinationen der Parameterwerte können nicht alle theoretisch möglichen Testfälle durchgeführt werden. Es stellt sich somit die Frage, ob Methoden existieren, die eine gezielte Reduktion der Testfälle bei einer bekannten Testabdeckung ermöglichen. Mögliche Lösungen dieser Forschungsfrage werden in Kapitel 8 thematisiert.



**Forschungsfrage 4**

*Kann die Simulation verwendet werden, um den Testbedarf an real durchgeführten Testkilometern zu reduzieren? Wie können dafür X-in-the-Loop-Verfahren eingesetzt werden? Welche Herausforderungen ergeben sich bei der Erstellung von verschiedenen Testtreibern, welche nach IEEE 610 (1990) eine zu testende Komponente/ein System aufruft und/oder steuert, für Elemente einer funktionalen Systemarchitektur von automatisierten Fahrfunktionen?*

Das vorherige Unterkapitel 1.2 hat hervorgehoben, dass eine Freigabe von automatisierten Fahrfunktionen mit Automatisierungsgrad 5, nach SAE (2014), durch Realtests im öffentlichen Straßenverkehr aus ökonomischen Gründen nicht möglich ist. Daher werden weitere Methoden der Testdurchführung benötigt, die den Bedarf an Realtests reduzieren. In diesem Zusammenhang wird oft die Simulation als mögliches Lösungsmittel genannt, da damit eine Vielzahl von Szenarien variiert und in kurzer Zeit gefahrenfrei und reproduzierbar getestet werden kann. Jedoch wird die Simulation kein Allheilmittel darstellen, da unter anderem, wie im vorherigen Unterkapitel 1.2 beschrieben, die Validierung und die Bewertung der Güte der Simulationsmodelle große Herausforderungen darstellt. Somit stellt sich die Frage, für welche Fragestellungen und Tests die Simulation oder ein X-in-the-Loop-Verfahren eingesetzt werden dürfen, um so die Anzahl der Realtests, die für eine Freigabe des Systems benötigt werden, zu reduzieren. Damit verbunden ist die Frage nach benötigten und realisierbaren Testtreibern für verschiedene Module einer funktionalen Systemarchitektur für automatisierte Fahrfunktionen. Testtreiber sind in diesem Fall Simulationsmodelle, die die Eingangsdaten für das Testobjekt zur Verfügung stellen und die Ausgangsdaten des Testobjekts verarbeiten. Eine detaillierte Darstellung von Testtreibern wird in Kapitel 9.4 gegeben. Durch die verschiedenen Automatisierungsgrade der Fahrfunktionen ergeben sich verschiedene Anforderungen an das Testkonzept und die Durchführung der Testfälle. Daher ist eine weitere Fragestellung, ob die Systeme mit verschiedenen Automatisierungsgraden in der Simulation getestet werden können und welche Herausforderungen für die Simulation und das Testkonzept resultieren. Eventuell ist ein Test dieser Systeme durch die verschiedenen Automatisierungsgrade nur in der Simulation möglich, da ein Realtest nicht möglich oder nicht gefahrenfrei durchführbar ist.

**Forschungsfrage 5**

*Wie können die Testergebnisse anhand von verschiedenen Kriterien bewertet werden und wie können die Einflüsse der Eingangsparameter anhand der Testergebnisse abgeschätzt werden?*

Diese Forschungsfrage beschäftigt sich mit der Bewertung und Analyse von Testfällen. Zum einen lautet eine Frage, wie die Testergebnisse anhand von verschiedenen Kriterien bewertet werden können. Zum anderen ist eine offene Fragestellung, ob und mit welchen Methoden die Einflüsse der Eingangsparameter auf die Bewertungskriterien anhand der Testergebnisse abgeschätzt werden können. Wenn diese Möglichkeit besteht, könnten aus den Testergebnissen neue Eingangsparameter für weiterführende Tests identifiziert werden. Damit wäre es möglich, die Parameterwerte zu finden, bei denen das Testobjekt am „besten“ oder am „schlechtesten“ abschneidet. Hierfür dürfen die Tests jedoch nicht nur auf *bestanden* und *nicht bestanden* überprüft werden, sondern wie „gut“ ein Test bestanden wurde. Somit wird eine Metrik für die Bewertung der Testergebnisse benötigt.

### Forschungsfrage 6

*Wie kann die Testabdeckung für automatisierte Fahrfunktionen gemessen und bewertet werden?*

Wie in Kapitel 1.2 dargelegt, handelt es sich bei automatisierten Fahrfunktionen um offene Systeme, die sich in einer offenen Umgebung bewegen. Dadurch kann nicht abgeschätzt werden, wie viele Szenarien insgesamt existieren und damit keine vollständige Testabdeckung nachgewiesen werden. Um die Fahrfunktionen dennoch freizugeben, werden Metriken benötigt, die die Testabdeckung messen beziehungsweise abschätzen können. Derzeit stellt die Messung der Testabdeckung eine offene Forschungsfrage dar, welche in der vorliegenden Arbeit nicht beantwortet und betrachtet wird. Dennoch wird die Forschungsfrage aus Gründen der Vollständigkeit an dieser Stelle formuliert.

Die einzige dem Autor bekannte Arbeit, die sich in diesem Umfang mit einem Gesamttestkonzept für automatisierte Fahrfunktionen beschäftigt hat, ist die Arbeit von Schmidt (2012), der ein szenariobasiertes Testkonzept für kamerabasierte Assistenzsysteme vorstellt, welches ebenfalls die Simulation nutzt. Schmidt (2012, S.11) beschreibt, dass die Thematik seiner Arbeit „verhältnismäßig“ neu ist und daher nur wenige wissenschaftliche Untersuchungen vorliegen. Die Arbeit von Schmidt (2012) wird in der vorliegenden Arbeit als Stand der Technik für ein Gesamttestkonzept betrachtet. Schmidt (2012, Kap.3) hat für seine Arbeit die folgenden Ziele definiert:

1. Ziel: Darstellung eines durchgängigen Gesamtkonzepts zur funktionalen Absicherung kamerabasierter aktiver Fahrerassistenzsysteme,
2. Ziel: Erarbeitung eines Hardware-in-the-Loop-Testsystems zum Testen kamerabasierter aktiver Fahrerassistenzsysteme,
3. Ziel: Definition eines Testfallbeschreibungsformats und einer Methode zur automatisierten Testfallgenerierung,
4. Ziel: Erarbeitung einer Methode zur Durchführung von Tests,
5. Ziel: Einbettung in bestehende Vorgehensweisen und Testprozesse,
6. Ziel: Berücksichtigen von Anwenderforderungen aus der industriellen Praxis.

Alle Ziele wurden nach Schmidt (2012, Kap.7) in seiner Arbeit erreicht. Im Folgenden werden die Ziele der Arbeit von Schmidt (2012) mit den hier gestellten Forschungsfragen verglichen. Das sechste Ziel wird nicht weiter betrachtet, da dieses Ziel die Anwenderforderungen an die von Schmidt (2012) entwickelte Simulationssoftware beschreibt und an dieser Stelle lediglich das Testkonzept verglichen wird. Durch den Vergleich der Ziele mit den hier gestellten Forschungsfragen wird überprüft, inwiefern die gestellten Forschungsfragen bereits beantwortet wurden oder ob weiterer Forschungsbedarf besteht.

**Ziel 1 und 5:** Das erste und fünfte Ziel von Schmidt (2012, S.86f, S.91f) werden gemeinsam betrachtet, da beide Ziele durch sein Rahmenkonzept erfüllt werden (Schmidt, 2012, S.134ff). Das Rahmenkonzept beinhaltet parametrierbare Testklassen, die Testplanung, die Testszenariogenerierung und die Umsetzung der Testfälle am Hardware-in-the-Loop (HiL) Teststand. Das Kernelement der Arbeit sind die parametrierbaren Testklassen und

die Umsetzung der Testfälle am HiL-Teststand. Die Testplanung und die Testszenariogenerierung werden von Schmidt (2012, S.134ff) nur peripher und beispielhaft betrachtet. Nach Schmidt (2012, S.134ff) sind die Testplanung und die Testszenariogenerierung jedoch „für das Gesamtkonzept essentiell [...] und in weiteren umfassenderen wissenschaftlichen Betrachtungen noch weiter [zu-]vertiefen [...]“ (Schmidt, 2012, S.134ff). Informationsquellen für die parametrierbaren Testklassen werden nicht betrachtet und analysiert. Damit wird von Schmidt (2012, Kap.5.1) die zweite Forschungsfrage weder gestellt noch beantwortet. Eine solche Analyse von möglichen Informationsquellen ist jedoch für die Erstellung von parametrierbaren Testklassen wichtig. Für die Testauswertung werden als Bewertungskriterien lediglich die Kriterien *bestanden* und *nicht bestanden* vorgeschlagen (Schmidt, 2012, S.137). Eine Metrik für die Erfüllung der gewünschten Testergebnisse wird somit nicht vorgeschlagen, wodurch die fünfte Forschungsfrage nicht beantwortet wird.

**Ziel 2:** Das zweite Ziel wurde nach Schmidt (2012) durch die Einführung eines „Visual Loop“ erfüllt. Damit wird ein geschlossener Testkreislauf beschrieben, welcher synthetische Bilder erzeugt und diese einem kamerabasierten Assistenzsystem über einen Bildschirm zur Verfügung stellt. Das Assistenzsystem nimmt die Bilder über eine Kamera wahr und sendet die Ausgangssignale an das HiL-System zurück.

Die Arbeit fokussiert sich stark auf kamerabasierte Assistenzsysteme, wodurch lediglich Bilder als Eingangsdaten für das System betrachtet werden. Die Erstellung von Eingangsdaten für weitere funktionale Systembestandteile, neben der Kamera, werden nicht betrachtet. Dies liegt unter anderem daran, dass keine funktionale Systemarchitektur des Assistenzsystems betrachtet wird. Damit können die Herausforderungen, die durch die Erstellung von Testtreibern für verschiedene Systembestandteile entstehen, nicht diskutiert werden. Dadurch wird die vierte Forschungsfrage nicht beantwortet.

**Ziel 3:** Das dritte Ziel wurde nach Schmidt (2012, Kap.5.3) durch die Definition des TESTE-Schemas und der Nutzung von kombinatorischen Algorithmen erfüllt. Das TESTE-Schema beschreibt für vordefinierte Szenariokacheln<sup>9</sup> die Elemente: Strecke (engl. track), Umweltbedingungen (engl. environment), Umgebung (engl. surrounding), Test-relevante Objekte (engl. test-relevant objects) und das Ego-Fahrzeug (engl. ego-vehicle) des Testfalls. Die kombinatorischen Algorithmen werden für die Anordnung der Kacheln verwendet.

Der von Schmidt (2012, Kap.5.3) vorgestellte Ansatz könnte eine Antwort auf die dritte gestellte Forschungsfrage geben. Jedoch nimmt das Ego-Fahrzeug durch die Beschreibungsform einen festen Bestandteil der Szenariobeschreibung ein. Dadurch können keine Szenarien beschrieben werden, in denen kooperative Funktionen, d.h. wenn die Testobjekte oder die Fahrfunktionen in mehreren Fahrzeugen gleichzeitig vorhanden sind, getestet werden. Weiterhin wird nicht beschrieben, wie das Verhalten der Test-relevanten Objekte definiert wird. Daher kann keine Aussage darüber getroffen werden, ob das Szenario reproduzierbar durchführbar ist, wenn die Testobjekte sich nicht deterministisch verhalten. Damit besteht weiterer Forschungsbedarf, um die dritte Forschungsfrage der vorliegenden Arbeit zu beantworten. Von Schmidt (2012, Kap.5.3.2) werden kombinatorische Algorithmen für die Anordnung der Kacheln verwendet. Die Algorithmen werden nicht verwendet, um die Parameter der parametrierbaren Testklassen zu variieren. Somit ist es nicht möglich, die Einflüsse der einzelnen Parameter auf das Testsystem zu analysieren, da durch die Kombinatorik nur

---

<sup>9</sup>Eine Szenariokachel beschreibt eine Vorlage für die Umgebung eines Szenarios mit quadratischen Abmaßen. Die Kacheln können dabei nahtlos aneinander gereiht werden.

ganze Kacheln ausgetauscht werden. Die Variation von Parametern durch kombinatorische Algorithmen wird von Schmidt (2012, S.236) als weiterer Forschungsgegenstand gesehen. Damit wird die dritte Forschungsfrage mit den Ansätzen von Schmidt (2012) nicht beantwortet. Eine weitere Betrachtung der parametrierbaren Testklassen und der kombinatorischen Testfallgenerierung von Schmidt (2012) findet im Kapitel 9.2 der vorliegenden Arbeit statt.

**Ziel 4:** Das vierte Ziel, die Erarbeitung einer Methode zur Durchführung von Tests, wird von Schmidt (2012, S.90f) durch den Aufbau eines HiL-Systems mit der Integration des Assistenzsystems inklusive einer realen Kamera als erfolgreich erfüllt angesehen.

In der Arbeit von Schmidt (2012, Kap.4.1.2) werden weitere X-in-the-Loop-Verfahren vorgestellt. Diese werden jedoch nicht im vorgestellten Testkonzept verwendet und integriert, da die Arbeit sich stark auf die Erstellung von synthetischen Bildern für den HiL-Prüfstand fokussiert. Damit findet keine Bewertung der Verwendbarkeit der verschiedenen X-in-the-Loop-Verfahren für unterschiedliche Tests einer automatisierten Fahrfunktionen statt. Schmidt (2012) trifft daher keine Aussage darüber, ob simulationsbasierte Tests einen Teil der real durchgeführten Tests ersetzen können. Damit wird die vierte Forschungsfrage durch das vierte Ziel von Schmidt (2012) nicht ausreichend beantwortet.

Auf den vorherigen Seiten wurde die Arbeit von Schmidt (2012) und die in der Arbeit verfolgten Ziele vorgestellt. Schmidt (2012) präsentiert in seiner Arbeit einen Entwurf für ein Testkonzept für kamerabasierte Assistenzfunktionen. Mit dem vorgestellten Testkonzept wird jedoch keine der aufgestellten Forschungsfragen ausreichend beantwortet. Aus diesem Grund reicht das Testkonzept von Schmidt (2012) nicht für den Nachweis der Sicherheit von automatisierten Fahrfunktionen, da unter anderem die Analyse und die Diskretisierung von wertkontinuierlichen Einflussparametern nicht thematisiert werden. Somit ergibt sich weiterhin ein Forschungsbedarf für ein generelles szenariobasiertes Testkonzept für automatisierte Fahrfunktionen, welches die aufgestellten Forschungsfragen beantwortet.

## 1.4 Aufbau der vorliegenden Arbeit

Die Arbeit ist in Kooperation mit der Volkswagen AG im Forschungsprojekt *UR:BAN*<sup>10</sup> (*Urbaner Raum: Benutzergerechte Assistenzsysteme und Netzmanagement*) entstanden. Das primäre Ziel zu Beginn der Arbeit war die Entwicklung einer Testumgebung für die entwickelte Assistenzfunktion des Engstellenassistenten (Scholz u. a., 2014). Während der Entwicklung wurde immer häufiger der Bedarf eines systematischen Testkonzepts für Assistenzsysteme und automatisierte Fahrfunktionen deutlich. Dadurch wurde der gewählte Ansatz in vielen Iterationen zum *modularen virtuellen Testbaukasten* weiterentwickelt, welcher ein generelles szenariobasiertes Testkonzept für automatisierte Fahrfunktionen darstellt. Somit hat sich der Fokus der Arbeit im Verlauf des Projekts verschoben.

Die vorliegende Arbeit unterteilt sich in fünf Teile. Der erste Teil beinhaltet die Einleitung mit der Motivation der vorliegenden Arbeit, dem Testdilemma von automatisierten Fahrfunktionen, der Identifikation des Forschungsbedarfs und dem Aufbau der Arbeit.

Der zweite Teil beschreibt die Grundlagen und Voraussetzungen. Dazu wird die Terminologie aus verschiedenen Bereichen beschrieben. Weiterhin werden Automatisierungsgrade für Fahrfunktionen und eine funktionale Systemarchitektur für automatisierte Fahrfunktionen

<sup>10</sup>Homepage des UR:BAN-Projekts: <http://www.urban-online.org>

vorgestellt. Der zweite Teil schließt mit einer Beschreibung von verschiedenen Methoden von Softwaretests und Verkehrssimulationen ab.

Der dritte Teil präsentiert den Aufbau und die Module des modularen virtuellen Testbaukastens als Prozessmodell für den systematischen Test von automatisierten Fahrfunktionen. Dazu werden Anforderungen an den Testbaukasten definiert, die durch die Forderung der Entwicklung eines effizienten, effektiven und systematischen Testkonzepts gestellt werden. Weiterhin werden Anforderungen beschrieben, die durch Anwender des Testkonzepts und durch die Testobjekte gestellt werden. Des Weiteren werden die Schnittstellen zwischen den Modulen des Testbaukastens präsentiert. In den folgenden Kapiteln werden die Module *Auswahl und Analyse von Einflussparametern*, *Testfallerstellung*, *Testdurchführung mittels X-in-the-Loop-Verfahren* und *Testfallauswertung* vorgestellt. Für jedes Modul des Testbaukastens werden Anforderungen definiert, ein ausgewählter Stand der Technik mit den gestellten Anforderungen verglichen und der Forschungsbedarf abgeleitet. Anschließend werden die Konzepte des Testbaukastens für das jeweilige Modul erläutert, um den Forschungsbeitrag darzustellen. Der dritte Teil schließt mit einer Beschreibung der Limitation des Testbaukastens ab, wobei die Erfüllung der gestellten Anforderungen beschrieben wird und die Limitation der verschiedenen Module des Testbaukastens dargestellt werden.

Der vierte Teil beschreibt die Anwendung des Testbaukastens im Projekt des Engstellenassistenten. Das Projekt wird vorgestellt und anschließend die verschiedenen Module des Testbaukastens anhand des Engstellenassistenten als Fallbeispiel für einen systematischen Test angewendet.

Der fünfte Teil fasst die Arbeit zusammen und stellt den zukünftigen Forschungsbedarf vor.



# Teil II

## Grundlagen und Voraussetzungen





## 2 Terminologie der vorliegenden Arbeit

Dieses Kapitel beschreibt die in dieser Arbeit verwendete Terminologie. Dabei werden allgemeine Begriffe, Begriffe aus dem Bereich des Testens und Begriffe zur Beschreibung von Schnittstellen erläutert.

### 2.1 Allgemeine Terminologie

Dieses Unterkapitel beschreibt eine allgemeine Terminologie für die vorliegende Arbeit. Es werden die Begriffe der Simulation und Emulation erläutert sowie der Unterschied zwischen der Effektivität und Effizienz dargestellt. Weiterhin werden verschiedene Skalen vorgestellt und der Begriff der Komplexität von Szenarien untersucht.

#### 2.1.1 Simulation und Emulation

Der Verein Deutscher Ingenieure (VDI) definiert den Begriff der Simulation allgemein als die „Nachbildung eines Systems mit seinen dynamischen Prozessen in einem experimentierfähigen Modell, um zu Erkenntnissen zu gelangen, die auf die Wirklichkeit übertragbar sind“ (VDI 3633, 2013). Ein Modell ist nach dem VDI „eine vereinfachte Nachbildung eines geplanten oder [bereits] real existierenden Originalsystems und[/oder] -prozesses in einem anderen begrifflichen oder gegenständlichen System. Es unterscheidet sich hinsichtlich der untersuchungsrelevanten Eigenschaften nur innerhalb eines vom Untersuchungsziel abhängigen Toleranzrahmens vom Bild“ (VDI 3633, 2013). Eine ähnliche Definition wird von Lackes u. a. (2016) vorgeschlagen. Die Simulation schafft mittels Abstraktion ein Modell zum Experimentieren, um die daraus resultierenden Ergebnisse anschließend auf das reale Problem anzuwenden. Nach Law und Kelton (1999, S.4f) ist die Simulation eine Ausführung eines mathematischen Modells, für das keine analytische Lösung existiert. Das mathematische Modell stellt dabei eine Abbildung des Zielsystems dar. Weiterhin wird die Simulation von Law und Kelton (1999, S.5f) unterteilt in deterministische oder stochastische, diskrete oder kontinuierliche Simulationen mit fixen oder variablen Zeitinkrementen.

Diese vorgestellte Definition und Kategorisierung der Simulation wird in dieser Arbeit verwendet. Jedoch wird der Begriff der Simulation vom Begriff der Emulation getrennt. Bei der Simulation (lateinisch *simulatio* = Vorspiegelung) wird ein System, wie schon oben definiert, durch ein mathematisches Modell beschrieben. Bei einer Emulation (lateinisch *aemulari* oder englisch *to emulate* = nachahmen) wird mittels einer Hardware, die der späteren Zielhardware ähnelt, das System nachgeahmt<sup>1</sup> (Funk, 2010). Damit können je nach Simulation oder Emulation des Systems unterschiedliche Berechnungsergebnisse vorliegen.

#### 2.1.2 Effektivität

Der Begriff der Effektivität wird in der ISO 9000 (2000, Nr. 3.2.14) definiert. Dort wird die Effektivität beschrieben als das „Ausmaß, in dem geplante Tätigkeiten verwirklicht und geplante Ergebnisse erreicht werden“. Thommen u. a. (2015) beschreiben die Effektivität mit

---

<sup>1</sup><https://www.kb.nl/en/organisation/research-expertise/research-on-digitisation-and-digital-preservation/emulation/what-is-emulation>, abgerufen: 15.02.2016

dem „Beurteilungskriterium, mit dem sich beschreiben lässt, ob eine Maßnahme geeignet ist, ein vorgegebenes Ziel zu erreichen“. Die Effektivität beschreibt somit das Verhältnis zwischen dem Ergebnis einer Maßnahme und dem Ziel, welches verfolgt wird. Die Effektivität ist größer eins, wenn das Ergebnis das erwartete Ziel übertrifft. Wenn das erreichte Ergebnis unter dem erwarteten Ziel liegt, ist die Effektivität kleiner eins. Für die erfolgreiche Umsetzung von Projekten sollte die Effektivität des Projekts größer oder gleich eins sein. Für den Bereich des Testens wird die Effektivität bestimmt durch die erreichte Testabdeckung und die Anzahl der gefundenen Fehler im Verhältnis zur Anzahl der durchgeführten Testfälle (Malz, 2013, S.288). Dabei findet eine effektive Testmethode bei einer gleichen Anzahl von durchgeführten Testfällen mehr Fehler als eine weniger effektive Testmethode. Howden (1978) vergleicht beispielsweise die Effektivität von verschiedenen strukturbasierten Testmethoden<sup>2</sup>.

### 2.1.3 Effizienz

Der Begriff der Effizienz wird in der ISO 9000 (2000, Nr. 3.2.15) definiert. Dort wird die Effizienz als das „Verhältnis zwischen dem erzielten Ergebnis und den eingesetzten Mitteln“ beschrieben. Feess u. a. (2015) beschreiben die Effizienz als ein „Beurteilungskriterium, mit dem sich beschreiben lässt, ob eine Maßnahme geeignet ist, ein vorgegebenes Ziel in einer bestimmten Art und Weise (z.B. unter Wahrung der Wirtschaftlichkeit) zu erreichen.“ Die Effizienz bewertet somit das Ergebnis einer Maßnahme im Verhältnis zum Aufwand, der für die Erreichung des Ergebnisses benötigt wurde. Die Effizienz ist eins, wenn das Ergebnis mit einem Aufwand erreicht wurde, welcher dem Ergebnis entspricht. Ist das Ergebnis größer als der Aufwand, ist die Effizienz dementsprechend größer. Wie das Ergebnis und der Aufwand im Projekt gemessen werden und welche Effizienz erreicht werden soll, sollte für jedes Projekt zu Beginn definiert werden. In der Informatik wird die Effizienz eines Algorithmus nach Ziegenbalg u. a. (2016, S. 155) anhand der benötigten Ressourcen, wie Speicherplatz (Speicher-Effizienz) oder Rechenzeit (Laufzeit-Effizienz), die für die Lösung eines Problems benötigt werden, gemessen. Für den Bereich des Testens wird die Effizienz, wie in der Informatik, bestimmt durch die benötigte Zeit und die eingesetzten Ressourcen für den Testprozess (Spillner und Linz, 2012, S.13).

### 2.1.4 Skalen

In der vorliegenden Arbeit werden verschiedene Skalen verwendet. Daher werden die *nominale* und *ordinale* Skala sowie die *Intervallskala* und die *Rationalskala* vorgestellt, wie sie in Eckstein (2014, S.23ff) beschrieben werden.

**Nominale Skala:** Auf einer nominalen Skala befinden sich Kategorien, die sich gegenseitig ausschließen. Die Kategorien haben dabei keine Ordnung oder Reihenfolge. Alle Kategorien haben damit die gleiche Wertigkeit. Ein Beispiel für eine nominale Skala wäre die Kategorie einer Straße mit den Werten: Autobahn, Landstraße oder Stadtstraße.

**Ordinale Skala:** Auf einer ordinalen Skala gibt es, wie bei einer nominalen Skala, sich ausschließende Kategorien. Im Gegensatz zu der nominalen Skala haben die Kategorien jedoch eine Rangordnung. Die Abstände zwischen den Kategorien müssen allerdings nicht gleich sein beziehungsweise sind diese gegebenenfalls unbekannt. Dies beeinflusst die Wahl der Analyseverfahren der Skala. Die Skala sagt lediglich aus, dass eine Kategorie vor oder

<sup>2</sup>Strukturbasierte Testmethoden nutzen Informationen über die innere Struktur des Testobjekts um Testfälle erstellen. Weitere Informationen sind unter anderem in Hoffmann (2008, S.201) zu finden.

nach einer anderen Kategorie angeordnet wird. Ein Beispiel wäre das Schulnotensystem mit den Stufen *sehr gut* bis *ungenügend*.

**Intervallskala:** Bei der Intervallskala werden, im Gegensatz zur ordinalen Skala, die Abstände zwischen den Skalenpunkten fest definiert. Voraussetzung ist dabei, dass die Skala auch dann noch gültig ist, wenn Punkte der Skala durch die Funktion  $f(x) = ax + b$  transformiert werden. Ein Beispiel ist der geometrische Abstand von Objekten. Der Abstand zwischen den Punkten 2m und 3m ist genauso groß, wie der Abstand zwischen den Punkten 102m und 103m, auch nach der Transformation durch die Funktion  $f(x)$ .

**Rationalskala:** Die Rationalskala hat, wie die Intervallskala, die Eigenschaft der gleichen Abstände der Skalenpunkte. Zusätzlich besitzt die Skala einen echten Nullpunkt. Hier muss die Transformationsgleichung  $f(x) = ax$  erfüllt sein. Mit der Rationalskala können Messwerte in Relation gesetzt werden. Beispielsweise besteht mit einer Rationalskala die Möglichkeit die Aussage zu treffen, dass eine Strecke  $a$  doppelt so lang ist wie eine Strecke  $b$ .

### 2.1.5 Komplexität

Der Begriff der Komplexität wird häufig bei der Beschreibung von Systemen oder Umgebungen verwendet. Dabei wird der Begriff oft dazu verwendet, um die Aussage zu treffen, dass ein System hochvernetzt ist oder das eine Umgebung für eine Fahrfunktion beziehungsweise für einen Menschen „schwer“ zu bewältigen ist. Eine Begründung für die Komplexität oder eine Metrik zur Bestimmung der Komplexität wird nicht gegeben. Daher wird folgend eine Auswahl von Literatur vorgestellt, die sich mit dem Begriff der Komplexität beschäftigt. Ziel ist die Identifikation von Dimensionen, mithilfe derer die Komplexität bestimmbar ist. Diese Dimensionen werden für eine Metrik zur Beschreibung der Komplexität von Szenarien und Szenen in verschiedenen Verkehrsdomänen, d.h. Autobahnen, Landstraßen oder Stadtstraßen, im Kontext von automatisierten Fahrfunktionen verwendet.

In der Komplexitätstheorie der Informatik wird die Komplexität über den Aufwand beziehungsweise die Ressourcen definiert, die für die Lösung einer bestimmten Aufgabe benötigt werden. Typische Komplexitätsgrade sind die Raum- oder Platzkomplexität, d.h. der Bedarf an Speicherplatz, oder die Zeitkomplexität, d.h. die Rechendauer für die Lösung des Algorithmus (Wegener, 2003, S.20ff, S.196ff). Um die Komplexität zu messen, wird die Implementierung eines Algorithmus auf die Arbeitsweise von einfachen Maschinen, deren Komplexität pro Arbeitsschritt bekannt ist, zurückgeführt.

In Meyers (2009) werden in einer Enzyklopädie Artikel über die Komplexität von Systemen über 35 Themengebiete zusammengefasst. Das Sammelwerk bietet damit einen domänenübergreifenden Überblick über die Verwendung des Begriffs der Komplexität. Zu Beginn jedes Artikels werden in einem Glossar die Begriffe des jeweiligen Artikels definiert. Die Definitionen der Begriffe der Komplexität oder des komplexen Systems sind über die Artikel hinweg nicht äquivalent. Jedoch lassen sich Gemeinsamkeiten und damit Dimensionen ableiten anhand derer die Komplexität bewertbar ist. In vielen Definitionen wird die Komplexität mit der Anzahl der Komponenten, die im System agieren, beschrieben. Eine weitere Dimension ist die Anzahl der Interaktionen zwischen den Bestandteilen des Systems, die im System auf verschiedene Arten vollführt werden können.

Schaub (1996) beschreibt den Begriff der Komplexität in Zusammenhang mit Situationen, bei denen Menschen besondere Schwierigkeiten haben Entscheidungen zu treffen. Nach Schaub (1996) sind ebenfalls die Anzahl der Elemente und deren Interaktionen, von ihm

*Vernetztheit* genannt, Dimensionen für die Komplexität. Zudem wird aufgeführt, dass nicht nur die Anzahl der Elemente entscheidend für die Komplexität ist, sondern auch die Anzahl der Zustände, die ein Element einnehmen kann. Nach Schaub (1996) lassen sich weitere Dimensionen für die Komplexität folgendermaßen zusammenfassen:

- Eigendynamik: Veränderung des Systems ohne direkte Eingriffe von außen
- Intransparenz: nicht vollständige Beobachtbarkeit des Systems und somit die Notwendigkeit von Indikatoren, um Informationen aus dem System zu extrahieren,
- Polytelie (Zielpluralität): Verfolgen von sich widersprechenden Zielen,
- Offenheit der Zielsituation: nicht eindeutig formulierte Ziele,
- Neuartig: Strukturen des Systems sind nicht bekannt und Hypothesenbildungen sind zur Exploration von bekannten Ergebnissen notwendig.

Feess (2015) beschreibt im *Gabler Wirtschaftslexikon* den Begriff der Komplexität für wirtschaftliche Systeme. Als Merkmale für die Komplexität werden von Feess (2015) ebenfalls die Anzahl und Art der Elemente sowie deren Beziehungen untereinander genannt. Des Weiteren wird die Eigendynamik und die Intransparenz des Systems als Merkmale für die Komplexität beschrieben. Nach Feess (2015) erfordert der Umgang mit komplexen Systemen ein hohes Maß an Wissen über die kausalen Zusammenhänge des Systems und die Fähigkeit, Komplexität auf wenige Merkmale zu reduzieren, um damit die Komplexität des Systems zu beherrschen.

Schweigert (2003, S.63f) beschreibt den Begriff der Komplexität in Bezug zum Straßenverkehr. Die Komplexität wird anhand des Vorhandenseins von entgegenkommenden und vorausfahrenden Fahrzeugen sowie von Kurven und Knotenpunkten gemessen. Um die Komplexität zu messen, wird ein Komplexitätsindex eingeführt. Für die Einflussfaktoren Kurven und Knotenpunkte wird für den Komplexitätsindex die Anzahl der Kurven und Knotenpunkte auf der gefahrenen Strecke zur Bestimmung der Komplexität herangezogen. Für den Einflussfaktor vorausfahrende Fahrzeuge „wird als Vergleichsmaßstab der mittlere zeitliche Anteil, bei dem im konkreten Streckenabschnitt auf vorausfahrende Fahrzeuge geblickt wurde, ermittelt“ (Schweigert, 2003, S.64).

Broggi u. a. (2015) kategorisieren die Komplexität von Szenarien für automatisierte Fahrzeuge anhand von fünf Dimensionen. Dabei wird unterschieden, ob ein Szenario *bekannt* oder *unbekannt* ist. Weiterhin wird zwischen *on-road* (Übersetzung: asphaltierte Straße) und *off-road* (Übersetzung: offenes Gelände) Straßen unterschieden. Die Geschwindigkeit des eigenen Fahrzeugs wird mit den Stufen *langsam* oder *schnell* bewertet. Die Umgebung wird als *kartografiert* oder *nicht kartografiert* beschrieben. Weiterhin kann die Umgebung *stationär*, *beweglich* und *vorhersagbar* oder *beweglich* und *nicht vorhersagbar* sein.

Die ausgewählte Literatur zeigt verschiedene Dimensionen zur Beschreibung von Komplexität. Die Übertragbarkeit der Komplexitätsmetrik aus der Informatik bezogen auf den Speicherplatz oder die Rechenzeit ist für die Beschreibung der Komplexität von Szenarien im Straßenverkehr nur bedingt möglich. Für die Beschreibung der Komplexität von Szenarien sind eher die Dimensionen, wie die Anzahl der Komponenten und deren Interaktionen, ausschlaggebend. Die Rechenzeit und der Speicherbedarf der Algorithmen können wiederum von der Anzahl der Komponenten im Szenario abhängen. Dies würde jedoch die Komplexität

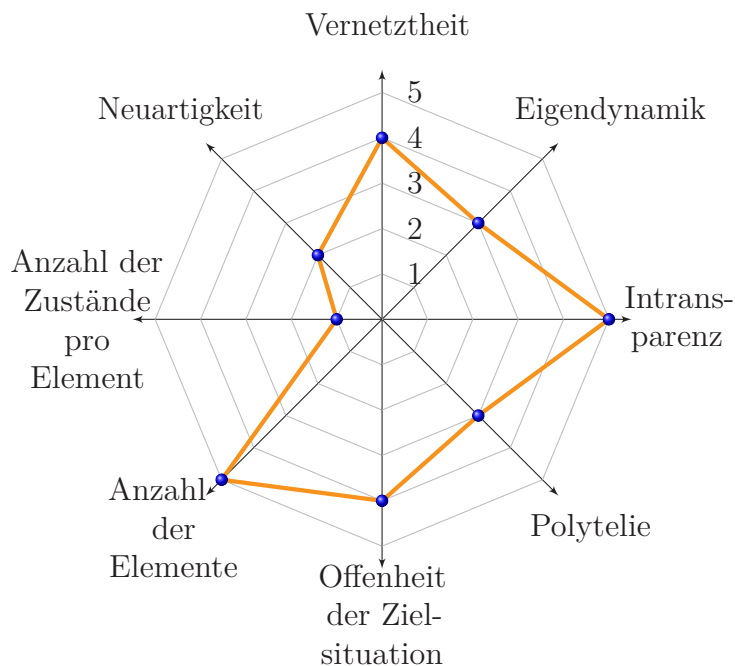
der Algorithmen bewerten und nicht die Komplexität des Szenarios. Daher wird der Komplexitätsbegriff der Informatik in der vorliegenden Arbeit nicht auf Szenarien im Straßenverkehr angewendet.

In den übrigen Definitionen wurde stets die Anzahl der Komponenten im System und deren Interaktionen als Dimensionen für die Komplexität eines Systems genannt. Weiterhin hat Schaub (1996) die Dimensionen Anzahl der Zustände, Offenheit der Zielsituation, Polytelie, Intransparenz, Eigendynamik und Neuartigkeit für die Komplexität von Entscheidungsprozessen vorgestellt. Automatisierte Fahrfunktionen führen ebenfalls eine Bewertung von Szenarien und Szenen durch, um Informationen für Entscheidungsprozesse zu generieren, was die in Kapitel 3.2 vorgestellte funktionale Systemarchitektur zeigt. In der vorliegenden Arbeit wird daher die Möglichkeit genutzt, die vorgeschlagenen Dimensionen für die Bewertung der Komplexität von Szenarien und Szenen zu verwenden. In Kapitel 2.3.5 werden die Dimensionen in einer Bewertung von Szenarien in den Straßendomänen Autobahn, Landstraße und Stadtstraße angewendet und damit die Übertragbarkeit der Dimensionen bewertet.

Durch die Vielzahl der beschreibenden Dimensionen wird deutlich, dass es sich bei der Komplexitätsbestimmung um ein multidimensionales Problem handelt, welches nicht ohne Hilfsmittel auf einen skalaren numerischen Wert zurückführbar ist. Für jede Anwendung sollte die Bedeutung der Dimensionen individuell bewertet werden. Aus diesem Grund wird in der vorliegenden Arbeit die Darstellung und Beschreibung der Komplexität von Szenarien und Szenen durch ein Kiviat-Diagramm (Liggesmeyer, 2009, S.248) vorgeschlagen. Nach Morris und Roth (1982), zitiert nach Hoffmann (2008, S.270f), ermöglichen Kiviat-Diagramme eine kombinierte Darstellung von mehrdimensionalen Skalen. Durch das Kiviat-Diagramm können weitere Dimensionen zur Bestimmung der Komplexität hinzugefügt werden, falls dies zu einem späteren Zeitpunkt notwendig ist. Die Achsen des Kiviat-Diagramms werden mit einer ordinalen Skala versehen, da zum einen nicht für alle Dimensionen ein absoluter Nullpunkt bekannt ist und zum anderen die Abstände der Kategorien auf den Skalen nicht bekannt sind. Daher wird die Skala auf folgende Art kodiert: Stufe 1 = sehr gering, Stufe 2 = gering, Stufe 3 = mittel, Stufe 4 = hoch und Stufe 5 = sehr hoch. Aufgrund der gewählten Skala können nur qualitative Aussagen über die Komplexität getroffen werden. Eine quantitative Aussage ist nur bedingt möglich. Durch eine Transformation der Skala auf die höherwertige Intervallskala besteht die Möglichkeit, einen Vergleich von Diagrammen unterschiedlicher Szenarien anhand des Medians durchzuführen (Kohn, 2005, S.15f). Bei der Transformation darf die Reihenfolge der Werte nicht verändert werden. Die vorgeschlagene Kodierung der Achsen stellt eine mögliche Transformation der Skalen dar.

Abbildung 2.1 zeigt ein ausgefülltes Kiviat-Diagramm mit den von Schaub (1996) vorgeschlagenen Achsen: Anzahl der Elemente, Anzahl von Zuständen pro Element, Neuartigkeit, Offenheit der Zielsituation, Polytelie, Intransparenz, Eigendynamik und Vernetztheit.





**Abbildung 2.1:** Kiviat-Diagramm zur Visualisierung der Komplexität von Szenarien mit den ordinalen Skalen und der gewählten Kodierung: 1 = sehr gering, 2 = gering, 3 = mittel, 4 = hoch und 5 = sehr hoch.

## 2.2 Begriffe des Testens

### 2.2.1 Testabdeckung

Der Begriff der Testabdeckung beschreibt das Verhältnis aus der Anzahl der durchgeführten Tests zu einem bestimmten Kriterium (Hamburg, 2015). Bei kontrollflussorientierten Testmethoden beschreibt die Testabdeckung den Test von Kontrollstrukturelementen. Hierfür sind Informationen über den inneren Aufbau des Testobjekts notwendig. Beispiele für Testabdeckungen bei kontrollflussorientierten Testmethoden sind die Anweisungsüberdeckung, die Zweigüberdeckung oder die Pfadüberdeckung (Liggesmeyer, 2009, S.41). Eine vollständige Testabdeckung bezogen auf diese Kriterien wird theoretisch durch erschöpfendes Testen mit White-Box Testmethoden (siehe Kapitel 4.1.1) erreicht. Praktisch bedeutet dies jedoch abhängig vom System einen erheblichen Aufwand (Spillner und Linz, 2012, S.14f). Die funktionsorientierten Testmethoden, welche die Funktionen des Testobjekts auf Basis der Eingangs- und Ausgangsdaten testen und damit zu den Black-Box Tests gehören (siehe Kapitel 4.1.1), erreichen aufgrund der unbekannten Struktur des Testobjekts nur einen eingeschränkten Rückschluss auf die erreichte Testabdeckung (Hoffmann, 2008, S.180). Die erreichte Testabdeckung wird bei diesen Testmethoden auf die getesteten Eingangsgrößen oder auf die getesteten Anforderungen bezogen (IEEE 610, 1990). Wenn alle Eingangsgrößen diskret sind, wird die Testabdeckung relativ zu den abgedeckten Kombinationen der Eingangswerte oder der abgedeckten Anforderungen berechnet. Sobald jedoch Eingangsparameter mit kontinuierlichen Wertebereichen vorhanden sind, ist aufgrund der notwendigen Diskretisierung keine vollständige Testabdeckung möglich (Kaiser, 2015b). Dies beschreibt noch einmal die in der Einleitung thematisierten Herausforderungen der vollständigen Testabdeckung für

automatisierte Fahrfunktionen. Sobald ein Eingangsparameter kontinuierliche Wertebereiche besitzt, wird keine allgemein gültige, vollständige Testabdeckung erreicht, da es theoretisch unendlich viele Testfälle gibt. Daher kann eine vollständige Testabdeckung nur bezogen auf ein definiertes Kriterium erreicht werden.

Daher wird in der vorliegenden Arbeit der Begriff der Testabdeckung auf die kombinatorische Testabdeckung für funktionsorientierte Testmethoden bezogen. Dabei wird davon ausgegangen, dass alle Eingangswerte bereits diskretisiert wurden. Die möglichen kombinatorischen Testabdeckungskriterien werden in Kapitel 8.4.3.1 beschrieben.

### 2.2.2 Testfall

Ein Testfall beschreibt die festgelegten Randbedingungen, Voraussetzungen zur Ausführung, Eingabewerte und die vorausgesagten Ausgaben oder das gewünschte Verhalten des Testobjekts (Spillner und Linz, 2012, S.10). Nach Liggesmeyer (2009, S.444) ist der Testfall die Betriebssituation, welche für die Testdurchführung verwendet wird. Hamburg (2015) unterscheidet dabei zwischen abstrakten und konkreten Testfällen. Von Spillner und Linz (2012, S.25) werden die abstrakten Testfälle auch logische Testfälle genannt. Konkrete Testfälle beinhalten konkrete Werte für die Eingangsparameter, während abstrakte Testfälle Platzhalter für die Parameterwerte vorhalten. Ein Platzhalter ist beispielsweise die Krümmung einer Kurve. Der Begriff des Testszenarios wird in dieser Arbeit bewusst nicht verwendet, um den Testfall von einem Szenario zu trennen. Ein Testfall ist in der vorliegenden Arbeit nach Abbildung 2.2 die Konkretisierung eines Szenarios, d.h. ein logisches Szenario inklusive gewählter Szenarioparameter. Weitere Bestandteile eines Testfalls sind die Bewertungskriterien, die entscheiden, unter welchen Bedingungen ein Testfall bestanden wurde, und Anforderungen an die Testdurchführung, wie die Anforderungen an die Genauigkeit, die nach DIN 5725 (2002) eine hohe Richtigkeit und Präzision voraussetzt, oder die benötigte Zeit zur Durchführung eines Testfalls.

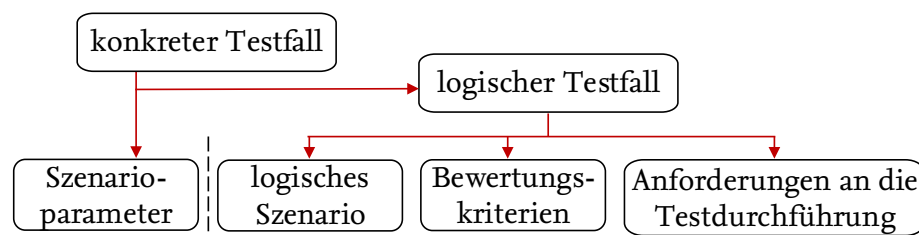


Abbildung 2.2: Bestandteile eines logischen und konkreten Testfalls

### 2.2.3 Verifikation

Nach Liggesmeyer (2009, S.518) ist die Verifikation die Überprüfung, ob die Ergebnisse der Entwicklung den Anforderungen zu Beginn des Projekts entsprechen. Die ISO 9000 (2000, Nr.3.8.4) beschreibt die Verifikation als Bereitstellung eines objektiven Nachweises, dass festgelegte Anforderungen erfüllt worden sind. Diese Definition wird auch von Sommerville (2006, S.804) und der ISO 26262 (2011, Teil 1, S.18) verwendet. Die Verifikation beschreibt damit allgemein die Erfüllung der Anforderungen durch das entwickelte System und beantwortet die Frage, ob das System *richtig* entwickelt wurde.

### 2.2.4 Validierung

Nach Sommerville (2006, S.804) beschreibt die Validierung einen Prozess, welcher überprüft, ob das entwickelte System den Erwartungen und Bedürfnissen des Kunden entspricht. Die ISO 9000 (2000, Nr.3.8.5) beschreibt die Validierung als „die Bestätigung durch einen objektiven Nachweis, dass die Anforderungen für eine bestimmte Anwendung oder einen bestimmten Gebrauch erfüllt sind“. Die Validierung beantwortet somit die Frage, ob die richtigen Anforderungen an das System gestellt wurden. Zum Vergleich zur Verifikation steht bei der Validierung die Frage im Fokus, ob das *richtige* System entwickelt wurde.

## 2.3 Begriffe zur Beschreibung von Schnittstellen

### 2.3.1 Szene

Ulbrich u. a. (2015) haben eine Literaturrecherche zur Definition des Begriffs der Szene durchgeführt und darauf basierend eine eigene Definition vorgeschlagen. Von Reschka (2016a, Kap.2) wurde die Definition angepasst und eine Unterscheidung zwischen beweglichen und unbeweglichen Elementen eingeführt. Die folgende angepasste Definition von Reschka (2016a, Kap.2) wird in der vorliegenden Arbeit verwendet.

#### Definition (Szene)

*Eine Szene beschreibt eine Momentaufnahme des Umfelds, welche die beweglichen und unbeweglichen Elemente des Umfelds, die Selbstrepräsentation aller Akteure und Beobachter und die Verknüpfung dieser Entitäten umfasst. Einzig eine Szenenrepräsentation in einer simulierten Welt kann allumfassend sein (objektive Szene). In der realen Welt ist sie immer unvollständig, fehlerbehaftet, unsicherheitsbehaftet und aus der Perspektive eines oder mehrerer Beobachter (subjektive Szene).*

Ein Akteur ist in dieser Definition ein selbst handelndes Element. Ein Beobachter ist ein wahrnehmendes Element innerhalb der Szene oder eines, das die Szene als ganzes betrachtet. Ein Element kann gleichzeitig Beobachter und Akteur sein. Bewegliche Elemente haben die Fähigkeit sich zu bewegen. Während der Bewegung sind die Elemente dynamisch, während sie im Stillstand statisch sind (Reschka, 2016a, S.59). Die Szenerie umfasst alle räumlich stationären Elemente.

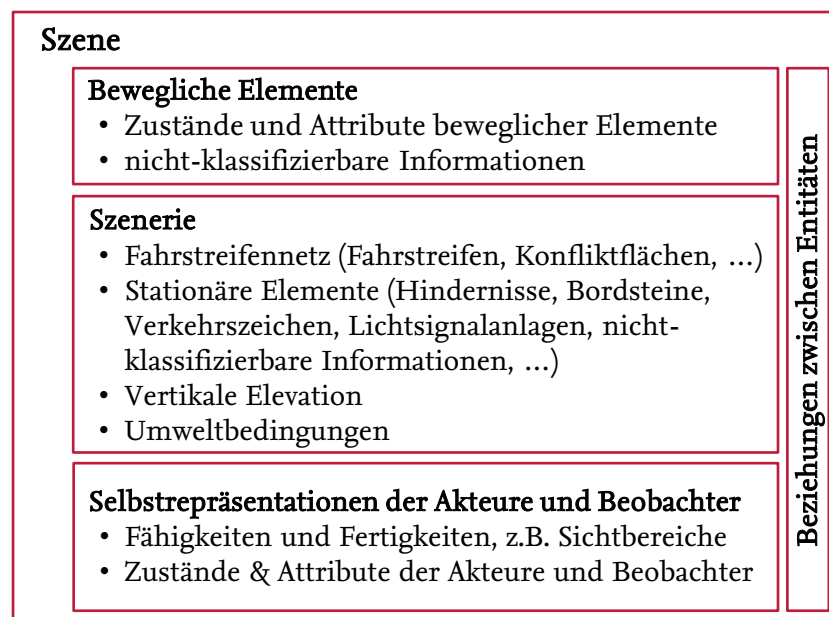
Die Bestandteile und Komponenten einer Szene sind in Abbildung 2.3 dargestellt. Eine Szene besteht aus der Szenerie, den beweglichen Elementen, den Selbstrepräsentationen aller Akteure und Beobachter sowie deren Beziehungen. Die Szenerie beschreibt alle unbewegliche Elemente und damit die räumlich stationären Aspekte einer Szene. Diese sind metrische, semantische und topologische Informationen, z.B. Fahrstreifen, Fahrstreifenmarkierungen, Fahrbahnoberflächen oder der Typ der Verkehrsdomäne. Darüber hinaus umfasst die Szenerie Informationen über Konfliktflächen zwischen Fahrstreifen und deren Verknüpfung untereinander, z.B. an Kreuzungen. Ebenso wie Matthaei (2015) subsumiert Abbildung 2.3 aus Sicht eines Fahrzeugs quasi-stationäre Wetter- und Lichtbedingungen als Teil der Szenerie. Auch zeitlich veränderliche Zustände von Lichtsignalanlagen und Wechselverkehrszeichen werden, da sie Attribute räumlich stationärer Elemente sind, als Teil der Szenerie verstanden. Diese Elemente können jedoch Akteure im Szenario darstellen.

Für bewegliche und unbewegliche Elemente müssen auch Informationen über nicht modellkonforme Elemente, beispielsweise aus einer Umfeldwahrnehmung resultierend, repräsentiert



werden. Dies können Informationen über nicht klassifizierbare, nicht verfolgbare oder nicht segmentierbare Messungen beziehungsweise nicht zum Entwicklungszeitpunkt bedachte Objektklassen sein. Das Vorhandensein solcher Elemente kann aus Sicht der funktionalen Sicherheit relevant für die Bewertung einer aus der Szene extrahierten Situation sein.

Die Szenenrepräsentation wird komplettiert durch die Selbstrepräsentationen der Akteure und Beobachter. Diese kann das aktuelle Fertigkeitenlevel, wie auch generelle Systemfähigkeiten und Fertigkeiten umfassen, welche in Reschka u. a. (2015a) detailliert beschrieben sind.



**Abbildung 2.3:** Bestandteile einer Szene (Ulbrich u. a., 2015) angepasst nach der Änderung von Reschka (2016a)

### 2.3.2 Situation

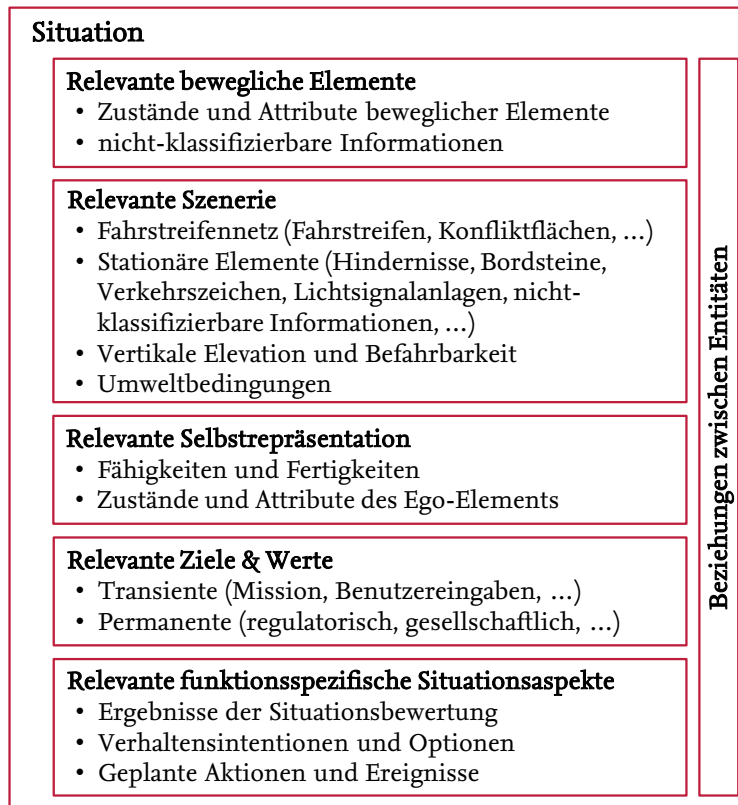
Ulbrich u. a. (2015) haben eine Literaturrecherche durchgeführt, um verschiedene Definitionen für den Begriff Situation zusammenzufassen. Analog zum Begriff der Szene wird für den Begriff der Situation eine Definition vorgeschlagen, die in dieser Arbeit verwendet wird.

#### Definition (Situation)

*Eine Situation beschreibt die Gesamtheit der Umstände, die für die Auswahl geeigneter Verhaltensmuster zu einem bestimmten Zeitpunkt zu berücksichtigen sind. Sie umfasst alle relevanten Bedingungen, Möglichkeiten und Determinanten von Handlungen. Eine Situation wird aus der Szene durch einen Prozess der Informationsauswahl und -augmentierung abgeleitet, basierend auf transienten (z.B. missionspezifischen) wie auch permanenten Zielen und Werten. Folglich ist eine Situation immer subjektiv, indem sie die Sicht eines Elements repräsentiert.*

Die Bestandteile und Komponenten einer Situation sind in Abbildung 2.4 dargestellt. Die Anzahl der Elemente einer Situationsrepräsentation weicht von denen der Szene durch eine ziel- und wertspezifische Informationsselektion und Augmentierung ab. Der wesentliche

Unterschied zwischen Szene und Situation ist, dass nur funktionsrelevante Informationen Teil der Situation sind. Abbildung 2.4 zeigt die Elemente einer Situationsrepräsentation.



**Abbildung 2.4:** Beispiel einer Situationsrepräsentation nach (Ulbrich u. a., 2015) angepasst an die Änderung von Reschka (2016a)

Unabhängig von den bereits für eine Szene diskutierten Elementen, umfasst eine Situation zusätzlich die funktionsrelevanten *Ziele und Werte*. Dies können transiente Ziele und Werte sein, wie die aktuelle Mission, Fahrmanöveranweisungen oder Verhaltenspräferenzen des Operators eines automatisierten Fahrzeugs. Bei einer teilautomatisierten Fahrfunktion können dies beispielsweise die Fahrmanöveranweisung zur Durchführung eines Fahrstreifenwechsels oder das Verändern einer Ziel-Zeitlücke in einer Folgefahrt sein. Solche Ziele und Werte können auch permanent sein, wie regulatorische oder gesellschaftliche Rahmenbedingungen. Aus Gründen der Laufzeiteffizienz kann es sinnvoll sein, nicht etwa jeden Paragraph einer Straßenverkehrsordnung alle paar Millisekunden in der Situation zu inkludieren, sondern diese Informationen in Form des aktuellen Landes und der Verkehrsdomäne (Autobahn, Landstraße, Stadt) zu abstrahieren.

### 2.3.3 Manöver

Für den Begriff des Manövers sind verschiedene Definitionen zu finden. Nach Dickmanns (2007, S.43) handelt es sich bei Manövern um eine charakteristische Zeithistorie von Steuergrößen, um den Übergang von einem Zustand eines Systems in einen anderen innerhalb eines Zeitraums zu erreichen. Schreiber (2012, S.57) unterscheidet zwischen expliziten und impliziten Manövern. Während das explizite Manöver vom Fahrer bestimmt wird, wird das

implizite Manöver von der Fahrfunktion selbst vorgegeben. Bei beiden Manöverarten werden Sollwerte für die Längs- und/oder Querführung definiert und somit eine Bewegung des Fahrzeugs beschrieben. Der Duden<sup>3</sup> und das Oxford Wörterbuch<sup>4</sup> beschreiben das Manöver zum einen als „große militärische Übung im Gelände, bei der Truppenbewegungen zweier gegnerischer Heere simuliert werden“ und zum anderen als „geschickt ausgeführte Wendung, taktische Bewegung (eines Truppenteils, Schiffes, Autos oder ähnliches)“.

Alle Definitionen beschreiben eine Zustandsänderung eines Systems. Im Duden und Oxford Wörterbuch sowie von Schreiber (2012) wird diese Zustandsänderung auf eine Bewegung des Systems oder der Systemkomponenten reduziert, wodurch diese auch Fahrmanöver genannt werden. Dies ist bei Dickmanns nicht der Fall, da dieser allgemein den Übergang von einem Zustand in einen anderen beschreibt.

In der vorliegenden Arbeit wird der Begriff des Manövers in Verbindung mit der Beschreibung und Regelung von Akteuren im Szenario verwendet. Dabei beschreibt ein Manöver mehr als eine Bewegung von Systemkomponenten. Durch ein Manöver wird ein Zustandswechsel des Szenarios beschrieben. Dies kann das Erscheinen von neuen Elementen oder die Zustandsänderung einer Lichtsignalanlage sein. Ein Manöver besitzt dabei eine oder mehrere Startbedingungen, Ereignisse und Aktionen. Die Ereignisse skizziert den Ablauf des Manövers, wobei jedes Ereignis durch eine oder mehrere Aktionen umgesetzt wird. Eine Aktion beschreibt beispielsweise eine Beschleunigung oder definiert das Abfahren einer Trajektorie durch einen Verkehrsteilnehmer. Weitere Details zur Beschreibung und Regelung der Akteure im Szenario durch den Testbaukasten werden im Kapitel 8.3.3 beschrieben.

### 2.3.4 Szenario

Der Begriff *Szenario* wird häufig im Kontext von Simulationen und Testen oder bei der funktionalen Beschreibung eines Systems verwendet. In Ulbrich u. a. (2015) wird dazu eine Literaturübersicht zusammengefasst. Ebenfalls wird eine Definition vorgeschlagen, die in der vorliegenden Arbeit verwendet wird.

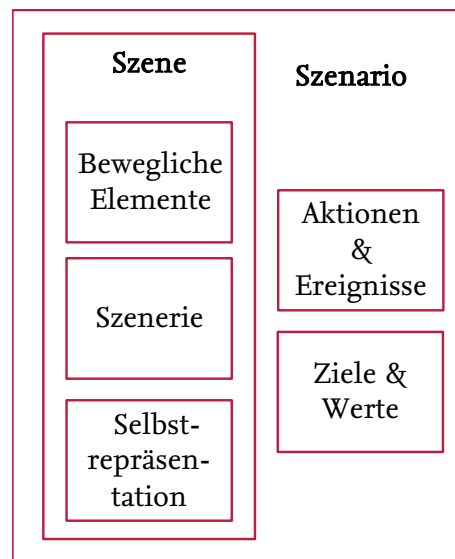
#### Definition (Szenario)

*Ein Szenario beschreibt die zeitliche Entwicklung von Szenenelementen innerhalb einer Folge von Szenen, welche mit einer Startszene beginnt. Aktionen und Ereignisse ebenso wie Ziele und Werte können spezifiziert werden, um diese zeitliche Entwicklung in einem Szenario festzulegen. Im Gegensatz zu Szenen decken Szenarien eine gewisse Zeitspanne ab.*

Gemäß Abbildung 2.5 umfasst ein Szenario Szenen, Aktionen und Ereignisse sowie Ziele und Werte. Ein Szenario verknüpft Szenen durch Aktionen und Ereignisse, die in der vorliegenden Arbeit in Manövern zusammengefasst werden. Es wird zwischen logischen und konkreten Szenarien unterschieden. Bei einem logischen Szenario werden Wertebereiche oder statistische Verteilungen für verschiedene Einflussparameter verwendet, während bei einem konkreten Szenario für jeden Parameter die Werte feststehen (Menzel, 2016). Gemäß Abbildung 2.6 beschreibt ein Szenario einen einzelnen Pfad in der zeitlichen Folge von Aktionen und Ereignissen (Kanten) sowie Szenen (Knoten) innerhalb des Baumes, welcher die Gesamtheit aller möglichen zukünftigen Szenarien für eine gegebene Ausgangsszene darstellt. Ein Szenario umfasst mindestens eine anfängliche Szene sowie Aktionen und

<sup>3</sup><http://www.duden.de/rechtschreibung/Manoeuver>, abgerufen: 16.02.2016

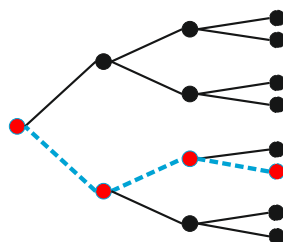
<sup>4</sup><http://www.oxforddictionaries.com/definition/english/manoeuvre?q=maneuver>, abgerufen: 16.02.2016



**Abbildung 2.5:** Bestandteile eines Szenarios (Ulbrich u. a., 2015)

Ereignisse, um einen Pfad in Abbildung 2.6 vollständig zu beschreiben. Es ist jedoch auch möglich, ein Szenario durch eine vollständige Menge von aufeinanderfolgenden Szenen zu spezifizieren. Dabei spezifizieren die Aktionen und Ereignisse nur noch das Verstreichen von Zeit bis zur nächsten Szene.

Entsprechend einer Theatermetapher wird ein Szenario typischerweise durch mehrere Szenen mit dazwischen vorgegebenen Aktionen und Ereignissen spezifiziert. In der Realität sind alle Aktionen und Ereignisse zu einem gewissen Grad mit Unsicherheiten bei deren Ausführung behaftet. Folglich passen menschliche Akteure ihr Verhalten an, um bestimmte Schlüsselszenen in einem Theaterspiel zu erreichen. Ebenso kann ein Verkehrsszenario solche Schlüsselszenen, wie eine eng definierte Durchfahrtreihenfolge in einem Kreuzungsszenario, enthalten. Im Extremfall ist ein Szenario durch ein Storyboard wie bei einem Zeichentrickfilm beschrieben, welches jedes kleinste Detail spezifiziert. Im Gegensatz dazu kann ein Szenario auch einzig durch eine Startszene und das Kommando an alle Akteure, ihre individuellen Ziele und Werte zu verfolgen, beschrieben sein, ohne dass weitere Szenen vorgegeben werden. Hierbei können jedoch möglicherweise vorherrschende stochastische Prozesse bei der Verhaltensausführung bereits nach wenigen Sekunden in einem komplett offenen Ausgang des Szenarios resultieren. Sind keine Formen von Unsicherheiten erlaubt, so resultieren beide Beschreibungsformen exakt im *gleichen* Szenario.



**Abbildung 2.6:** Ein Szenario (blau gestrichelt) als zeitliche Abfolge von Aktionen/Ereignissen (Kanten) und Szenen (Knoten) (Ulbrich u. a., 2015)

### 2.3.5 Metrik für die Komplexität von Szenarien

Nachdem im vorherigen Unterkapitel der Begriff des Szenarios vorgestellt wurde, wird in diesem Unterkapitel der Ansatz zur Definition einer Metrik für die Komplexität von Szenarien beschrieben. Ziel der Metrik ist die qualitative Bestimmung der Komplexität von Szenarien für die Straßendomänen Autobahn, Landstraße und Stadtstraße. Dazu wird die in Kapitel 2.1.5 eingeführte Beschreibung des Begriffs der Komplexität mit den von Schaub (1996) vorgestellten Dimensionen *Intransparenz*, *Eigendynamik*, *Vernetztheit*, *Neuartigkeit*, *Anzahl der Zustände pro Element*, *Anzahl der Elemente*, *Offenheit der Zielsituation*, *Polytelie* verwendet.

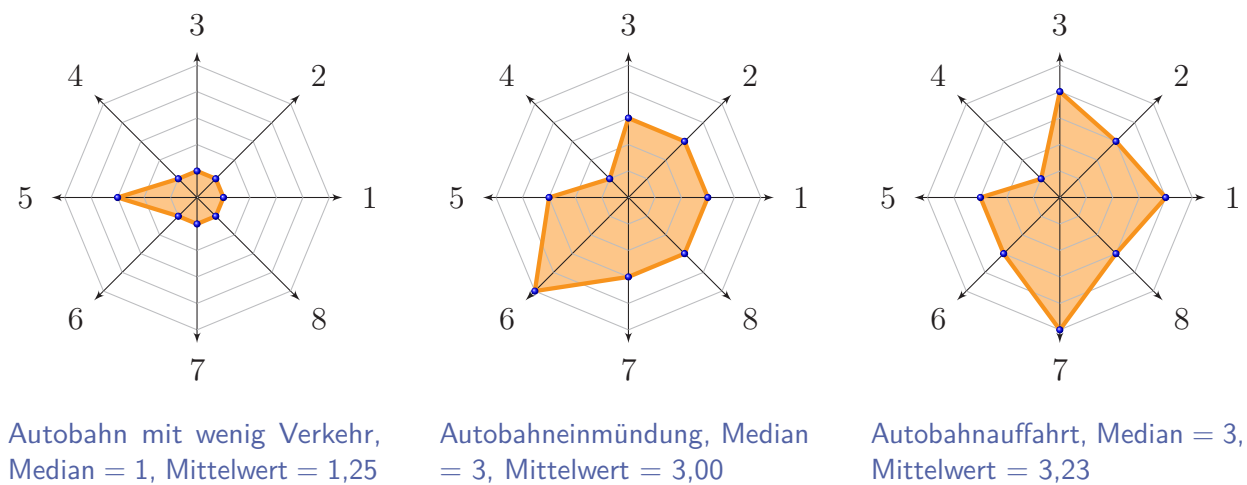
Bubb u. a. (2015, S.130) haben die Komplexität von jeweils drei Szenarien auf der Autobahn, der Landstraße und der Stadtstraße auf einer ordinalen Skala mit den Stufen *gering*, *mittel* und *hoch* eingliedert. Im Folgenden wird analysiert, ob die gewählten Einstufungen der Szenarien durch die in Kapitel 2.1.5 vorgestellte Definition zur Komplexität bestätigt und daraus ein Ansatz einer Metrik zur Bestimmung der Komplexität ableitbar ist. Die Achsen der Kiviat-Diagramme werden mit einer ordinalen Skala versehen und auf folgende Art kodiert: Stufe 1 = sehr gering, Stufe 2 = gering, Stufe 3 = mittel, Stufe 4 = hoch und Stufe 5 = sehr hoch. Durch diese Transformation der ordinalen Skala auf die höherwertige Intervallskala, bei der die Reihenfolge der Werte durch die Transformation nicht verändert wird, darf ein Vergleich der Kiviat-Diagramme anhand des Medians durchgeführt werden (Kohn, 2005, S.15f). Weiterhin ist die Reihenfolge der Komplexität der Szenarien durch eine Dominanz unter den Szenarien bestimmbar. Von einer Dominanz wird gesprochen, wenn ein Szenario im Vergleich zu den übrigen Szenarien in jeder Dimension mindestens den gleichen oder einen höheren Wert auf den Skalen besitzt. Neben dem Median wird der Mittelwert als Vergleichswert angegeben. Mit diesem darf aus Sicht der Statistik jedoch nicht gerechnet werden, da der Abstand zwischen den Skalenpunkten aufgrund der ordinalen Skala unbekannt ist.

Die Verteilung der Punkte auf der ordinalen Skala basiert als ersten Ansatz auf einer intuitiven Einschätzung des Autors der vorliegenden Arbeit, die auf der Beschreibung der Szenarien von Bubb u. a. (2015, S.130) beruht. Die Beschreibungen von Bubb u. a. (2015, S.130) beschränken sich auf wenige Worte, sodass die getroffene Einschätzung eine rein subjektive Bewertung ermöglicht. Dies wird nochmals in der Zusammenfassung dieses Kapitels thematisiert. Die Skalenwerte werden relativ zwischen den verschiedenen Szenarien gewählt. Im Folgenden werden die Einträge im Kiviat-Diagramm für die drei Szenarien je Straßendomäne erläutert. Dabei wird für jede Straßendomäne auf jeder Dimension eine Reihenfolge beschrieben, um sie der ordinalen Skala zuzuordnen. Die Anzahl der Zustände der übrigen Elemente im Szenario werden nicht bewertet, da davon ausgegangen wird, dass diese über alle Szenarien konstant ist. Als Elemente werden im Folgenden die übrigen Verkehrsteilnehmer verstanden. Weiterhin wird die Neuartigkeit als konstant auf eine Stufe definiert, da die Neuartigkeit vom System abhängt und auf Basis der Szenariobeschreibung nicht bewertbar ist.

#### Autobahn

Auf der Autobahn hat nach Bubb u. a. (2015, S.130) ein Szenario eine geringe Komplexität, wenn wenig Verkehr vorhanden ist. Der Autor der vorliegenden Arbeit versteht darunter eine gerade Autobahn ohne Ausfahrten mit wenig Verkehr. Als Beispiel für mittlere Kom-

plexität wird eine Autobahneinmündung genannt. Eine Autobahneinmündung wird in der vorliegenden Arbeit als ein Zusammenschluss von zwei Autobahnen verstanden, bei der die Verkehrsteilnehmer die Autobahn nicht verlassen können. Für eine hohe Komplexität wird das Beispiel einer Autobahnauffahrt herangezogen. Abbildung 2.7 zeigt die qualitative Einteilung der von Bubb u. a. (2015, S.130) beschriebenen Szenarien im Kiviat-Diagramm zur Komplexität von Szenarien basierend auf der subjektiven Bewertung. Im Folgenden werden die Einteilungen der Dimensionen erläutert.



**Abbildung 2.7:** Kiviat-Diagramme für die verschiedenen Komplexitätsgrade der Autobahn. Legende: 1: Intransparenz, 2: Eigendynamik, 3: Vernetztheit, 4: Neuartigkeit, 5: Anzahl der Zustände pro Element, 6: Anzahl der Elemente, 7: Offenheit der Zielsituation, 8: Polytellie

**Intransparenz:** Die Intransparenz steigt kontinuierlich über die drei Szenarien. Auf einer Autobahn mit wenig Verkehr wird gegenüber den übrigen Szenarien eine sehr geringe Intransparenz geschätzt, da alle Verkehrsteilnehmer das Ziel haben den Fahrstreifen zu folgen. Beim Szenario der Autobahneinmündung wird von einer mittleren Intransparenz gegenüber den übrigen beiden Szenarien ausgegangen. Die Ziele der übrigen Verkehrsteilnehmer sind bekannt, da die Autobahn an der Autobahneinmündungen nicht verlassen werden kann und somit alle Verkehrsteilnehmer weiterhin das Ziel verfolgen, den Fahrstreifen zu folgen. Jedoch kann das Zusammenführen von zwei Autobahnen zu einem erhöhten Verkehrsaufkommen führen und dadurch zu vermehrten Fahrstreifenwechseln. Dadurch wird in diesem Szenario die Intransparenz auf die Stufe mittel geschätzt. Bei der Autobahnauffahrt wird von einer hohen Intransparenz ausgegangen, da die Auffahrt zu vermehrten Fahrstreifenwechseln führt und zudem zu beschleunigenden und verzögernden Verkehrsteilnehmern, die die Autobahn verlassen oder auf diese auffahren.

**Eigendynamik:** Für das Szenario mit wenig Verkehr wird eine geringe Eigendynamik im Vergleich zu den übrigen Szenarien geschätzt. Dies wird durch das geringe Verkehrsaufkommen und die unveränderte Anzahl der Verkehrsteilnehmer im Szenario begründet. Bei der Autobahnauffahrt und -einmündung kann sich die Anzahl der Verkehrsteilnehmer auf der Autobahn ändern, wodurch sich das System auch ohne direkte Eingriffe stärker verändert und damit eine höhere Eigendynamik besitzt. Die Eigendynamik der beiden übrigen Szenarien wird als gleichwertig auf die Stufe mittel geschätzt.



**Vernetztheit:** Die Vernetztheit wird beim Szenario mit wenig Verkehr als sehr gering vermutet, da die wenigen Verkehrsteilnehmer dem Straßenverlauf folgen und das Ziel verfolgen, den Fahrstreifen zu halten. Damit ergibt sich eine geringe Interaktion zwischen den Verkehrsteilnehmern. Bei der Autobahneinmündung ergibt sich eine mittlere Vernetztheit zwischen den Verkehrsteilnehmern, da dort Verkehrsteilnehmer vereinzelt den Fahrstreifen wechseln, um schneller im Verkehr zu fahren. Allerdings verfolgen alle Verkehrsteilnehmer weiterhin das Ziel den Fahrstreifen zu folgen. Für die Autobahnauffahrt ergibt sich eine hohe Vernetztheit, da dort Verkehrsteilnehmer auf die Autobahn auffahren oder die Autobahn verlassen. Für das Verlassen der Autobahn oder das Auffahren auf diese ist eine hohe Interaktion zwischen den Verkehrsteilnehmern notwendig, da gegebenenfalls Verkehrsteilnehmer auf der Autobahn Platz für auffahrende Verkehrsteilnehmer zur Verfügung stellen müssen.

**Anzahl der Elemente:** Die Anzahl der Elemente steigt vom ersten zum zweiten Szenario an. Beim Szenario mit wenig Verkehr wird von einer sehr geringen Anzahl von Verkehrsteilnehmern ausgegangen. Für die Autobahneinmündung wird angenommen, dass durch den Zusammenschluss von zwei Autobahnen die Anzahl von Verkehrsteilnehmern größer als in den beiden übrigen Szenarien ist. Daher wird die Anzahl der Elemente auf sehr groß geschätzt. Bei der Autobahnauffahrt wird von einer mittleren Anzahl von Verkehrsteilnehmern ausgegangen, da der Zu- und Abfluss durch die Autobahnauffahrt im Verhältnis zur Autobahneinmündung nicht so groß ist.

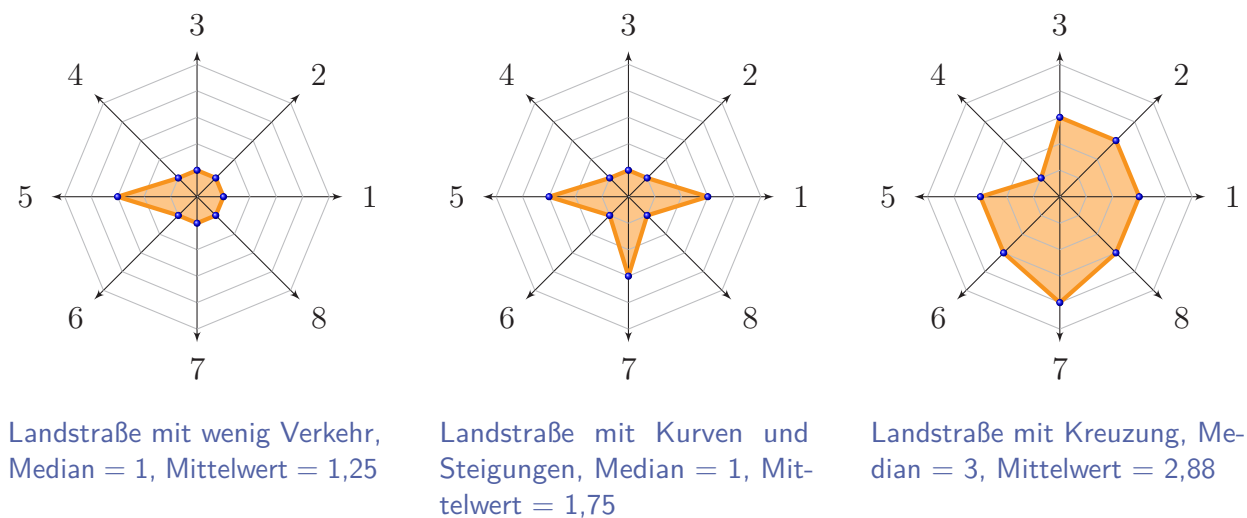
**Offenheit der Zielsituation:** Die Offenheit der Zielsituation wird beim ersten Szenario als sehr gering eingeschätzt, da alle Verkehrsteilnehmer das gleiche Ziel verfolgen und keine Verkehrsteilnehmer die Autobahn verlassen oder auf diese auffahren. Für die Autobahneinmündung wird von einer mittleren Offenheit der Zielsituation ausgegangen, da vermutet wird, dass Verkehrsteilnehmer aufgrund der höheren Anzahl von Verkehrsteilnehmern die Fahrstreifen wechseln, um schneller im Verkehr zu fahren. Weiterhin verfolgen alle Verkehrsteilnehmer das gleiche Ziel. Für das Szenario der Autobahnauffahrt wird von einer sehr hohen Offenheit der Zielsituation ausgegangen, da Verkehrsteilnehmer die Autobahn verlassen und neue Verkehrsteilnehmer auf die Autobahn auffahren und deren Ziele nicht bekannt sind.

**Polytelie:** Die Polytelie ist schwer abschätzbar, da diese direkt von den Zielen der Verkehrsteilnehmer anhängig ist. Beim ersten Szenario wird jedoch geschätzt, dass die Polytelie sehr gering ist, da nur das Ziel des Fahrstreifenfolgens vorhanden ist. Bei der Autobahneinmündung und der Autobahnauffahrt wird eine höhere Polytelie vermutet, da es durch das Verhalten der übrigen Verkehrsteilnehmer zu Zielkonflikten kommen kann. Ein Beispiel wäre eine Szene mit einem Verkehrsteilnehmer, der auf die Autobahn auffährt, und einem anderen Verkehrsteilnehmer, der gleichzeitig die Autobahn verlässt. Daher wird beim zweiten und dritten Szenario von einer mittleren Polytelie ausgegangen.

Bei der Betrachtung der Mediane der ersten beiden Szenarien zeigt sich, dass die von Bubb u. a. (2015, S.130) beschriebene Reihenfolge der Szenarien bezüglich der Komplexität bestätigt wird. Das Szenario der Autobahnauffahrt hat den gleichen Median wie die Autobahneinmündung, wodurch das Szenario nicht als komplexer eingestuft wird. Bei der Betrachtung der Mittelwerte ergibt sich die gleiche Reihenfolge, die von Bubb u. a. (2015, S.130) beschrieben wird.

## Landstraße

Ein Szenario auf der Landstraße hat nach Bubb u. a. (2015, S.130) eine geringe Komplexität, wenn wenig Verkehr vorhanden ist. Der Autor der vorliegenden Arbeit versteht darunter eine gerade Landstraße mit wenig Verkehr. Als Beispiel für eine mittlere Komplexität wird eine Landstraße mit Kurven und Steigungen genannt. Ein Beispiel für hohe Komplexität wird von Bubb u. a. (2015) nicht genannt. Daher wird vom Autor der vorliegenden Arbeit für ein Szenario mit hoher Komplexität eine Kreuzung definiert, bei der der Verkehr durch eine Lichtsignalanlage geregelt wird. Abbildung 2.8 zeigt die qualitative Einteilung der von Bubb u. a. (2015, S.130) beschriebenen Szenarien in den Kiviat-Diagrammen nach der subjektiven Einschätzung. Im Folgenden werden die Einteilungen der Dimensionen erläutert.



**Abbildung 2.8:** Kiviat-Diagramme für die verschiedenen Komplexitätsgrade der Landstraße. Legende: 1: Intransparenz, 2: Eigendynamik, 3: Vernetztheit, 4: Neuartigkeit, 5: Anzahl der Zustände pro Element, 6: Anzahl der Elemente, 7: Offenheit der Zielsituation, 8: Polytelie

**Intransparenz:** Aufgrund des angenommenen geringen Verkehrsaufkommens im ersten Szenario wird die Intransparenz als sehr gering eingestuft. Beim Szenario mit Kurven und Steigungen wird eine mittlere Intransparenz vermutet, da die übrigen Verkehrsteilnehmer durch die Kurven und Steigungen später wahrgenommen werden. Die gleiche Intransparenz wird für das Kreuzungsszenario geschätzt, da dort beispielsweise die korrekten Fahrstreifen für die Überquerung der Kreuzung gewählt werden müssen. Dabei werden die Fahrstreifen und die Kreuzungstopologie gegebenenfalls erst spät wahrgenommen. Da der Verkehr durch eine Lichtsignalanlage gesteuert wird, müssen keine Vorfahrtsregeln erkannt und angewendet werden, wodurch sich eine mittlere Intransparenz ergibt.

**Eigendynamik:** Aufgrund des angenommenen geringen Verkehrsaufkommens wird davon ausgegangen, dass sich das System nicht stark ändert. Daher wird die Eigendynamik für die ersten beiden Szenarien auf sehr gering geschätzt. Weiterhin haben weder Kurven noch Steigungen einen großen Einfluss auf die Eigendynamik des Szenarios. Beim Kreuzungsszenario wird eine mittlere Eigendynamik geschätzt, da dort Verkehrsteilnehmer die Fahrtrichtung ändern und sich dadurch das Szenario auch ohne direkte Eingriffe ändert. Durch die Lichtsignalanlage wird die Eigendynamik der Verkehrsteilnehmer und damit die Eigendynamik des



Szenarios gesteuert und eingeschränkt, wodurch eine mittlere Eigendynamik abgeschätzt wird.

**Vernetztheit:** Die Vernetztheit der Elemente wird für die ersten beiden Szenarien auf sehr gering geschätzt. Aufgrund des geringen Verkehrsaufkommens wird von einer sehr geringen Interaktion und damit Vernetztheit zwischen den Verkehrsteilnehmern ausgegangen. Beim Szenario mit Kurven und Steigungen wird die Vernetztheit der Verkehrsteilnehmer auf sehr gering geschätzt, da die Kurven und Steigungen keinen Einfluss auf die Vernetztheit der Verkehrsteilnehmer haben. Beim dritten Szenario wird von einer mittleren Vernetztheit ausgegangen. Durch eventuell vorhandene Konfliktflächen in der Kreuzung müssen die Verkehrsteilnehmer in Interaktionen treten, um die Konflikte zu lösen und die Kreuzung zu überqueren. Durch die Lichtsignalanlage wird ein Teil der Interaktion zwischen den Verkehrsteilnehmern vorgegeben, sodass von einer mittleren Vernetztheit ausgegangen wird.

**Anzahl der Elemente:** In den ersten beiden Szenarien wird von einer sehr geringen Anzahl von Elementen ausgegangen, da das Szenario mit wenigen Verkehrsteilnehmern beschrieben wird. Beim zweiten Szenario wurde keine Aussage über die übrigen Verkehrsteilnehmer getroffen. Da die Komplexitätssteigerung von Bubb u. a. (2015, S.130) durch die Kurven und Steigungen beschrieben wird, wird davon ausgegangen, dass ebenfalls wenig Verkehrsteilnehmer vorhanden sind. Beim dritten Szenario wird von einer mittleren Anzahl von Verkehrsteilnehmern gegenüber den ersten beiden Szenarien ausgegangen, da an einer Kreuzung potentiell mehr Verkehrsteilnehmer vorkommen als auf einer einzelnen Landstraße.

**Offenheit der Zielsituation:** Die Offenheit der Zielsituation wird beim ersten Szenario auf sehr gering geschätzt, da alle Verkehrsteilnehmer das Ziel des Fahrstreifenfolgens verfolgen. Weiterhin befinden sich nur wenige Verkehrsteilnehmer auf der Landstraße. Beim Szenario mit Kurven und Steigungen wird von einer mittleren Offenheit der Zielsituation ausgegangen, da die übrigen Verkehrsteilnehmer womöglich erst spät wahrgenommen werden. Beim Kreuzungsszenario wird eine hohe Offenheit der Zielsituation geschätzt. Durch die Kreuzung treffen sich Verkehrsteilnehmer mit verschiedenen Zielen an einem Ort. Da die Ziele der einzelnen Verkehrsteilnehmer nicht bekannt sind, ergibt sich in diesem Szenario eine hohe Offenheit der Zielsituation.

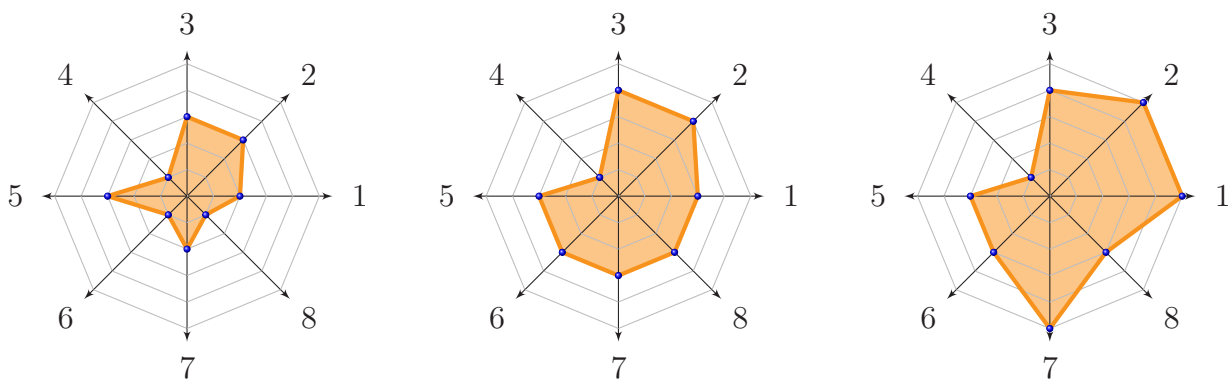
**Polytelie:** Die Polytelie ist, wie bei den Autobahnszenarien, schwer abschätzbar, da diese wiederum direkt von den Zielen der Verkehrsteilnehmer abhängig ist. Es wird jedoch vermutet, dass die Polytelie in den ersten beiden Szenarien sehr gering ist, da nach der Szenariobeschreibung nur das Ziel des Fahrstreifenfolgens vorhanden ist. Beim Kreuzungsszenario wird die Polytelie auf die mittlere Stufe geschätzt, da beispielsweise durch die verschiedenen möglichen Fahrtrichtungen an einer Kreuzung und die sich ergebenden Konfliktflächen verschiedene Ziele ergeben.

Für die ersten beiden Szenarien ergibt sich der gleiche Wert für den Median. Daher wird durch den Median keine Reihenfolge der Szenarien bezüglich der Komplexität festgelegt. Das zweite Szenario erreicht in allen Dimensionen mindestens die gleichen Skalenwerte, wie das erste Szenario. Bei den Dimensionen der Intransparenz und der Offenheit der Zielsituation wird jeweils ein höherer Skalenwert erreicht, sodass sich eine Dominanz ergibt und das zweite Szenario als komplexer bewertet wird. Der Mittelwert zeigt die gleiche Reihenfolge, wie von Bubb u. a. (2015, S.130) beschrieben. Das Kreuzungsszenario hat einen Median von 3 und

ist damit komplexer als die beiden übrigen Szenarien, wodurch die beschriebene Reihenfolge von Bubb u. a. (2015, S.130) bestätigt wird.

## Stadtstraße

Für Bubb u. a. (2015, S.130) hat ein innerstädtisches Szenario eine geringe Komplexität, wenn wenig Verkehr vorhanden ist und die Straßen knotenfrei sind. Als Beispiel für mittlere Komplexität wird eine Stadt mit beschilderten Kreuzungen genannt, bei der der Verkehrsfluss von Lichtsignalanlagen gesteuert wird. Weiterhin werden Engstellen und Kurven als Szenarien mit mittlerer Komplexität gesehen. Eine hohe Komplexität hat ein städtisches Szenario, welches beschilderte Kreuzungen mit Wartepflicht beinhaltet. Abbildung 2.9 zeigt die qualitative Einteilung der beschriebenen Szenarien von Bubb u. a. (2015, S.130) auf die Dimensionen der Kiviat-Diagramme nach der subjektiven Einschätzung. Im Folgenden werden die Einteilungen der Dimensionen erläutert.



Stadt mit wenig Verkehr, Median = 2, Mittelwert = 2,00

Stadt mit beschilderten durch Lichtsignalanlagen gesteuerte Kreuzungen, Median = 3, Mittelwert = 3,00

Stadt mit beschilderten Kreuzungen und Wartepflicht, Median = 3,5, Mittelwert = 3,63

**Abbildung 2.9:** Kiviat-Diagramme für die verschiedenen Komplexitätsgrade der Stadt. Legende: 1: Intransparenz, 2: Eigendynamik, 3: Vernetztheit, 4: Neuartigkeit, 5: Anzahl der Zustände pro Element, 6: Anzahl der Elemente, 7: Offenheit der Zielsituation, 8: Polytelie

**Intransparenz:** Beim ersten Szenario wird eine geringe Intransparenz geschätzt. Dies wird, wie schon bei beiden vorherigen Szenarien auf der Autobahn und Landstraße, mit der geringen Anzahl an Verkehrsteilnehmern begründet. Weiterhin wird die Straße als knotenfrei beschrieben, wodurch keine Kreuzungen vorhanden sind. Für die Szenarien Kreuzung mit Lichtsignalanlage, Engstelle und Kurven wird die Intransparenz auf mittel geschätzt, da dort die übrigen Verkehrsteilnehmer erst spät wahrgenommen werden und das Verhalten insbesondere an Engstellen und Kreuzungen nicht vollständig vorhersehbar ist. Beim Kreuzungsszenario mit Wartepflicht wird die Intransparenz auf sehr hoch geschätzt, da dort keine Infrastrukturelemente, wie Lichtsignalanlagen, den Verkehr steuern und damit die Intransparenz senken.

**Eigendynamik:** Die Eigendynamik wird für das erste Szenario auf die Stufe mittel geschätzt. Durch die geringe Anzahl von Verkehrsteilnehmern ist die Eigendynamik gegenüber

den beiden anderen Szenarien als geringer einzuschätzen. In Relation zu den Szenarien auf der Autobahn und Landstraße wird vermutet, dass aufgrund der vielfältigeren Verkehrsteilnehmer, neben Fahrzeugen auch Radfahrer und Fußgänger, eine höhere Eigendynamik vorhanden ist. Beim Kreuzungsszenario mit Lichtsignalanlage wird die Eigendynamik auf hoch eingeschätzt, da dort Verkehrsteilnehmer die Fahrtrichtung ändern und sich dadurch das Szenario auch ohne direkte Eingriffe verändert. Aufgrund der vielfältigeren Verkehrsteilnehmer ergibt sich im Vergleich zum Kreuzungsszenario auf der Landstraße ein höherer Skalenwert. Beim Kreuzungsszenario mit Wartepflicht wird eine sehr hohe Eigendynamik geschätzt. Dies liegt an der fehlenden Infrastrukturunterstützung zur Steuerung des Verkehrsflusses. Dadurch haben die Verkehrsteilnehmer eine höhere Eigendynamik, da sich ihr Verhalten nur nach grundlegenden Verkehrsregeln richtet und sie damit größere Freiheiten haben.

**Vernetztheit:** Für das erste Szenario wird eine mittlere Vernetztheit geschätzt. Trotz der geringen Anzahl an Verkehrsteilnehmern wird eine mittlere Vernetztheit vermutet, da die Verkehrsteilnehmer in einem engeren Verkehrsraum miteinander agieren. Damit wird eine höhere Vernetztheit im Vergleich zu den Szenarien mit wenig Verkehr auf der Landstraße und der Autobahn geschätzt. Beim Kreuzungsszenario mit einer Lichtsignalanlage wird die Vernetztheit auf der Stufe hoch geschätzt. Dies wird, wie beim Kreuzungsszenario auf der Landstraße, mit potentiell vorhandenen Konfliktflächen begründet. Der Verkehrsfluss wird durch eine Lichtsignalanlage gesteuert, sodass die Vernetztheit durch die Lichtsignalanlage im Verhältnis zum Kreuzungsszenario mit Wartepflicht reduziert wird. Beim Kreuzungsszenario mit Wartepflicht wird dementsprechend die Vernetztheit auf sehr hoch geschätzt.

**Anzahl der Elemente:** Im ersten Szenario wird von einer sehr geringen Anzahl von Verkehrsteilnehmern ausgegangen, da das Szenario mit wenig Verkehr beschrieben wird. Beim zweiten und dritten Szenario wird von einer mittleren Anzahl von Verkehrsteilnehmern gegenüber dem ersten Szenarien ausgegangen, da an Kreuzungen potentiell mehr Verkehrsteilnehmer vorkommen.

**Offenheit der Zielsituation:** Die Offenheit der Zielsituation wird beim ersten Szenario als gering eingeschätzt. Dies wird durch die Beschreibung der Straße, als knotenfrei, begründet. Dadurch folgen die Verkehrsteilnehmer lediglich dem Fahrstreifenverlauf, wodurch sie die Fahrtrichtung nicht ändern. Beim zweiten und dritten Szenario besteht die Möglichkeit, dass die übrigen Verkehrsteilnehmer durch die Kreuzung die Fahrtrichtung ändern. Damit ergibt sich eine höhere Offenheit der Zielsituation, da nicht bekannt ist, welcher Verkehrsteilnehmer in welche Richtung fährt. Daher wird beim Kreuzungsszenario mit Lichtsignalanlage die Offenheit der Zielsituation auf mittel geschätzt. Beim Kreuzungsszenario mit Wartepflicht wird eine sehr hohe Offenheit der Zielsituation vermutet. Der Unterschied wird durch die Lichtsignalanlage begründet, da diese den Verkehrsfluss aktiv steuert und damit die Offenheit der Zielsituation beeinflusst. Beim Szenario mit Wartepflicht wird eine größere Offenheit der Zielsituation geschätzt, da der Verkehr nicht aktiv durch das Infrastrukturelement gesteuert wird.

**Polytelie:** Die Polytelie ist, wie bei den Autobahn- und Landstraßenszenarien, schwer abschätzbar, da diese direkt von den Zielen der Verkehrsteilnehmer abhängig ist. Die Polytelie wird jedoch für das erste Szenario als gering geschätzt, da durch die Beschreibung der Straße, als knotenfrei, nur das Ziel des Fahrstreifenfolgens vorhanden ist. Für das Kreuzungsszenario

mit Lichtsignalanlage und mit Wartepflicht wird die Polytelie auf die Stufe mittel geschätzt. Durch die möglichen Fahrtrichtungen an der Kreuzung und die potentiell vorhandenen Konfliktflächen ergeben sich verschiedene Ziele für die Verkehrsteilnehmer.

Der Median der drei Einteilungen zeigt, dass die Reihenfolge der Komplexität von Bubb u. a. (2015, S.130) bestätigt wird. Die größere Komplexität des zweiten Szenarios zeigt sich unter anderem durch die höhere Intransparenz, die höhere Anzahl der Elemente und die höhere Polytelie. Das Szenario der beschilderten Kreuzungen mit Wartepflicht zeigt die höchste Intransparenz und Eigendynamik, da die Verkehrsteilnehmer Verkehrsregeln wie „rechts vor links“ beachten müssen und der Verkehr nicht durch Lichtsignalanlagen gesteuert wird. Dies spiegelt sich ebenfalls in einer Offenheit der Zielsituation wieder.

### 2.3.6 Zusammenfassung

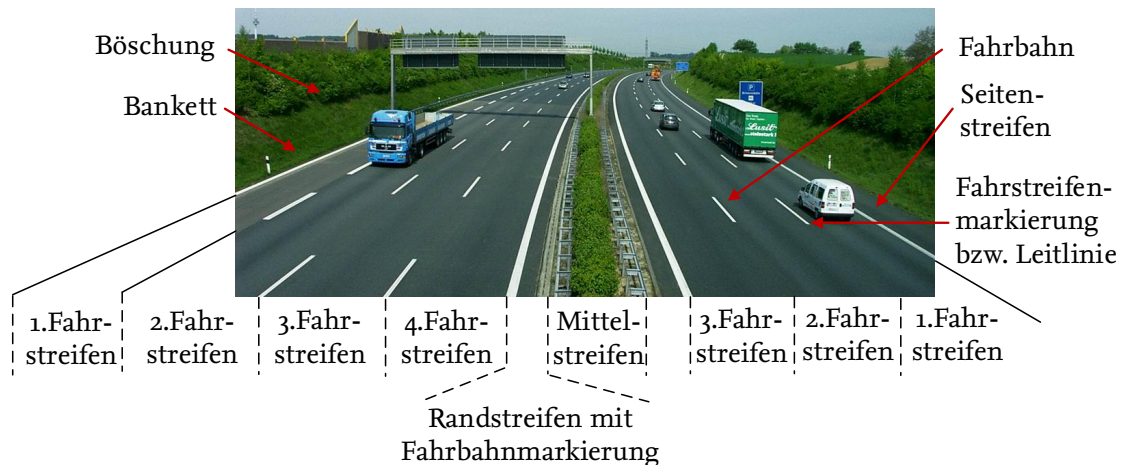
Mit den gewählten Dimensionen im Kiviat-Diagramm wurde gezeigt, dass ein qualitativer Vergleich der Komplexität von Szenarien möglich ist und die von Schaub (1996) vorgeschlagenen Dimensionen zur Bewertung der Komplexität auf automatisierte Fahrfunktionen übertragbar sind. Die von Bubb u. a. (2015, S.130) gewählte Reihenfolge der Komplexität der verschiedenen Szenarien in den Domänen Autobahnen, Landstraßen und Stadtstraßen wurde bestätigt. Jedoch wird aufgrund der ordinalen Skala lediglich eine Aussage zur relativen Reihenfolge anhand des Medians getroffen. Ein absoluter Wert für die Komplexität wird nicht bestimmt. Mit Hilfe der Kiviat-Diagramme werden die Ursprünge der Komplexität identifizierbar. Damit wird gezeigt, welche der Dimensionen der Diagramme von einer automatisierten Fahrfunktion besonders „gut“ beherrscht werden sollte, um ein Szenario mit einer bestimmten Komplexität zu beherrschen.

Das Ziel dieses Kapitels war die Herleitung einer allgemeinen Metrik für die Bestimmung der Komplexität von Szenarien. Für eine Metrik wird nach Winner (2016) eine Skala und eine Messvorschrift für diese Skala benötigt. Eine Skala wurde durch die ordinale Skala und die Dimensionen im Kiviat-Diagramm hergeleitet. Für die Herleitung einer Metrik fehlt eine allgemein gültige Messvorschrift, da derzeit die Verteilung der Punkte auf den Skalen intuitiv aus der Beschreibung der Szenarien gewählt wird. Daher findet eine subjektive Bewertung statt, die sich je nach Anwender verändert. Somit werden weitere Forschungsarbeiten benötigt, um Messvorschriften für die gewählte Skala zu finden, um daraus eine allgemein gültige Metrik für die Komplexität von Szenarien zu definieren. Mit dem in der vorliegenden Arbeit vorgestellten Ansatz ist derzeit lediglich eine qualitative, subjektive Bewertung der Komplexität möglich.

## 2.4 Begriffe des Straßenaufbaus

In dieser Arbeit werden Begriffe des Aufbaus einer Straße verwendet. Folgend wird in diesem Abschnitt der grundsätzliche laterale und longitudinale Aufbau einer Straße erläutert. Dazu werden Definitionen für die grundlegenden Elemente einer Straße vorgestellt. Abbildung 2.10 zeigt den exemplarischen Aufbau einer Autobahn in Deutschland. Die Autobahn besteht aus zwei Fahrbahnen, wobei für jede Fahrtrichtung eine Fahrbahn vorgesehen ist. Die Fahrbahn besteht wiederum aus den Fahrbahnmarkierungen, die auf den Randstreifen aufgebracht sind, und den Fahrstreifen sowie deren Fahrstreifenmarkierungen. Die Seitenstreifen gehören nicht zur Fahrbahn. Die Fahrbahnen werden nach der Forschungsgesellschaft für Straßen-

und Verkehrswesen (FGSV, 2008, S.18ff) baulich getrennt. Im Beispiel der Autobahn wird die bauliche Trennung durch den Mittelstreifen realisiert.



**Abbildung 2.10:** Schematischer Aufbau einer Autobahn mit Regelquerschnitt 36<sup>5</sup>

Der Regelquerschnitt beschreibt den lateralen Aufbau der Straße und die verschiedenen Breiten der Fahrstreifen. Tabelle 2.1 gibt einen Überblick über verschiedene Regelquerschnitte für Autobahnen und Landstraßen. Zu sehen ist jeweils die Beschreibung der linken Fahrbahn. Für die rechte Fahrbahn gelten die gleichen Breiten gespiegelt am Mittelstreifen. Ausnahmen sind die Regelquerschnitte *RQ15* und *RQ15,5*, da hier die rechte und linke Fahrbahn wechselseitig einen beziehungsweise zwei Fahrstreifen haben. Der zweite temporäre Fahrstreifen ist in der Tabelle mit einem Stern markiert. Im Folgenden werden die Aufgaben der jeweiligen Bestandteile der Straße erläutert.

Auf dem Mittelstreifen werden die Stützpfeiler der Überführungsbauwerke, Fahrzeugrückhaltesysteme und Verkehrszeichen platziert. Weiterhin dient der Mittelstreifen zur Entwässerung der Fahrbahn (FGSV, 2008, S.20). Die Randstreifen trennen die Fahrbahn auf der einen Seite vom Mittelstreifen und auf der anderen Seite vom Seitenstreifen. Bei Landstraßen ist der Randstreifen nur bei einigen Regelquerschnitten vorhanden, siehe Tabelle 2.1. Auf dem Randstreifen werden die Markierungen des Fahrbahnrandes aufgebracht. Auf Autobahnen haben die Randstreifen die Aufgabe die Sichtweite in Linkskurven zu erhöhen. Des Weiteren wird durch den Randstreifen beim Nothalten am Mittelfahrstreifen den Fahrzeuginsassen das Aussteigen erleichtert (FGSV, 2008, S.19).

Die Fahrstreifen werden getrennt durch die Leitlinien (Verkehrszeichen VZ340) oder die Fahrbahnmarkierungen (Verkehrszeichen VZ295). Die Forschungsgesellschaft für Straßen- und Verkehrswesen (FGSV, 1980, S.6ff) definiert verschiedene Typen und Breiten der Fahrstreifenmarkierungen. Die Breite eines Fahrstreifens wird bestimmt durch die Entfernung zwischen dem inneren Rand der Fahrbahnmarkierung und der Mitte der Leitlinie oder als Abstand von zwei Leitlinien.

Der Seitenstreifen beziehungsweise Standstreifen liegt zwischen der Fahrbahn und dem Bankett. Bei Landstraßen ist der Seitenstreifen genauso wie der Randstreifen nur bei bestimmten Regelquerschnitten vorhanden. Der Seitenstreifen wird aus Gründen der Verkehrssicherheit,

<sup>5</sup>Bildquelle: <https://commons.wikimedia.org/wiki/File:A8palmbach.jpg>, abgerufen 15.06.2016



N&R	Name	EK	B in m	Sei- tens. in m	Rs in m	Fahrstreifen in m				Rs in m	Ms in m
						1.	2.	3.	4.		
RAL	RQ9	EKL4	1,5		0,50	2,50					
RAL	RQ11	EKL3	1,5		0,50	3,50					
RAL	RQ11,5	EKL2	1,5		0,50	3,50					0,5
RAL	RQ15	EKL1	1,5		0,75	3,50	3,25*				0,5
RAL	RQ15,5	EKL1	1,5		0,75	3,50	3,25*				1,0
RAL	RQ21	EKL1	1,5		0,50	3,50	3,25			0,50	2,5
RAA	RQ25	EKA3	1,5	2,0	0,50	3,50	3,25			0,50	2,5
RAA	RQ28	EKA2	1,5	2,5	0,75	3,50	3,50			0,75	3,0
RAA	RQ31	EKA1	1,5	3,0	0,75	3,75	3,75			0,75	4,0
RAA	RQ31,5	EKA3	1,5	2,0	0,50	3,50	3,25	3,25		0,50	2,5
RAA	RQ36	EKA1	1,5	2,5	0,50	3,75	3,50	3,50		0,75	4,0
RAA	RQ38,5	EKA3	1,5	2,0	0,50	3,50	3,50	3,25	3,25	0,50	2,5
RAA	RQ43,5	EKA1	1,5	2,5	0,50	3,75	3,75	3,50	3,50	0,75	4,0

**Tabelle 2.1:** Übersicht der Regelquerschnitte (RQ) für Autobahnen nach den Richtlinien für die Anlage von Landstraßen (RAL) und Autobahnen (RAA), Abkürzungen: N&R: Norm und Richtlinie, B: Böschung, Seitens.: Seitenstreifen, Rs: Randstreifen, Ms: Mittelstreifen, EK: Entwurfsklasse, EKL: Entwurfsklasse Landstraße, EKA: Entwurfsklasse Autobahn (FGSV, 2008, 2013)

zum Beispiel zum Abstellen von defekten Fahrzeugen, oder für die Erweiterung der Straße für die Einrichtung von Arbeitsstellen verwendet (BMVI, 2009).

Das Bankett dient, wie der Mittelstreifen, zur Unterbringung von Fahrzeugrückhalteinrichtungen, Leiteinrichtungen oder Verkehrszeichen. Dabei stellt das Bankett einen nicht befestigten Bereich der Straße dar und trennt den Seitenstreifen von der Böschung. Die Funktion des Banketts ist die Entwässerung der Straße sowie die Befestigung des Seitenstreifens (FGSV, 2008, S.19).

Die Böschung ist der äußerste Teil der Straße, welcher Höhenunterschiede zwischen der Straße und dem übrigen Gelände ausgleicht. Es wird unterschieden zwischen der *Einschnittsböschung*, wenn das Bankett unterhalb des Geländes liegt, und der *Dammböschung*, wenn das Bankett oberhalb des Geländes liegt (FGSV, 2008, S.20f).

Je nach prognostizierter Verkehrsstärke für den jeweiligen Streckenabschnitten werden verschiedene Entwurfsklassen und Regelquerschnitte durch die Normen und Richtlinien (FGSV, 2008) vorgeschlagen. Für die Autobahnen gibt es die Entwurfsklassen *EKA1*: Fernautobahn beziehungsweise Überregionalautobahn, *EKA2*: autobahnähnliche Straße sowie *EKA3*: Stadtautobahn. Bei Straßen der Klasse *EKA1* handelt es sich um Bundesautobahnen, für die nach der Forschungsgesellschaft für Straßen- und Verkehrswesen (FGSV, 2008, S.22) je nach Verkehrsstärke ein Regelquerschnitt von *RQ31*, *RQ36* oder *RQ43,5* empfohlen wird. Die übrige Zuteilung der Regelquerschnitte für die weiteren Entwurfsklassen der Autobahnen und Landstraßen sind der Tabelle 2.1 zu entnehmen.

Für Stadtstraßen gibt es ebenfalls Entwurfsklassen für den Aufbau der Straße und die Breiten der Fahrstreifen. Jedoch ist für die Entscheidung über den Querschnitt von Straßen in Städten eine vielschichtige Entscheidung notwendig, die sich über einen Entscheidungsbaum, der sich wiederum über verschiedene Kriterien aufspaltet, darstellen lässt. Die weitere Betrachtung für Straßen in Städten wird in der vorliegenden Arbeit nicht weiter fortgeführt. Informationen über den Entscheidungsbaum und die Regelquerschnitte für Stadtstraßen werden von der Forschungsgesellschaft für Straßen- und Verkehrswesen (FGSV, 2006) beschrieben.





## 3 Automatisierte Fahrfunktionen

Das folgende Kapitel beschreibt die Automatisierungsgrade von Fahrfunktionen. Dabei werden mögliche Umsetzungen der automatisierten Fahrfunktionen in den jeweiligen Automatisierungsgraden präsentiert. Des Weiteren wird eine funktionale Systemarchitektur vorgestellt, die eine Umsetzung von Fahrfunktionen in verschiedenen Automatisierungsgraden ermöglicht. Weiterhin wird eine Fahrfunktion vorgestellt, an der die exemplarische Anwendung des Testbaukastens im Laufe der vorliegenden Arbeit gezeigt wird.

### 3.1 Funktionale Beschreibung von Automatisierungsgraden

Sowohl die *Society of Automotive Engineers (SAE)* (SAE, 2014) als auch die *Bundesanstalt für Straßenwesen (BASt)* (Gasser u. a., 2012) haben Automatisierungsgrade für Fahrfunktionen definiert, die sich zum Teil in der Namensgebung unterscheiden. Der *Verband der Automobilindustrie e. V.* hat in einem Strategiepapier die Terminologie von Gasser u. a. (2012) übernommen und um den Automatisierungsgrad 5 - fahrerlos erweitert (VDA, 2015). Bei der Vorstellung der Automatisierungsgrade wird jeweils auf die Definition der SAE sowie auf die Definition der BASt eingegangen. Die Automatisierungsgrade werden bezüglich der Übernahme der Fahraufgabe in der Längs- und/oder Querführung des Fahrzeugs, der Aufgabe der Überwachung des Systems und der Fahrumgebung, der Rückfallebene und der Leistungsfähigkeit der Fahrfunktion unterschieden. Ziel der Definitionen ist eine einheitliche Beschreibung der Automatisierungsgrade, um verschiedene Fahrfunktionen in Gruppen zusammenfassen, die in den obengenannten Kategorien ähnlichen Ausprägungsstufen aufweisen. Die Automatisierungsgrade werden vorgestellt, um in Kapitel 5.3 Anforderungen zu identifizieren, welche durch die verschiedenen Automatisierungsgrade an den modularen virtuellen Testbaukasten gestellt werden. Im Folgenden werden die verschiedenen Automatisierungsgrade erläutert und vorhandene Systeme vorgestellt, die den verschiedenen Automatisierungsgraden entsprechen.

#### **Automatisierungsgrad 0 - Keine Automation**

Beim Automatisierungsgrad 0 findet keine Automation durch eine Fahrfunktion statt. Dies bedeutet, dass der Fahrer über die komplette Zeit die Längs- und Querführung des Fahrzeugs übernehmen muss. Die Aufgabe der Überwachung des Fahrzeugs und der Umwelt liegt beim Fahrer. Die Rückfallebene ist ebenfalls der Fahrer. Die BASt definiert die Stufe als „driver only“.

#### **Automatisierungsgrad 1 - Fahrerassistenz**

Beim Automatisierungsgrad Fahrerassistenz wird entweder die Längs- oder die Querführung des Fahrzeugs von einer automatisierten Fahrfunktion übernommen, wodurch der Fahrer bei diesem Teil der Fahraufgabe entlastet wird. Der Fahrer hat jedoch die Aufgabe, die Assistenzfunktion dauerhaft zu überwachen. Den jeweils anderen Teil der Fahrzeugführung muss der Fahrer dauerhaft während der Fahrt übernehmen. Die Rückfallebene für das System stellt ebenfalls der Fahrer dar.

Ein Längsführungssystem mit Automatisierungsgrad 1, welches sich in Serie befindet, ist der Abstandsregeltempomat (engl. Adaptive Cruise Control (ACC), Full Speed Range ACC) (ISO 22179, 2009; ISO 15622, 2010). Das System übernimmt dauerhaft die Längsführung des Fahrzeugs und hält dabei eine Wunschgeschwindigkeit. Sollte die Einhaltung der Wunschgeschwindigkeit aufgrund eines vorausfahrenden Verkehrsteilnehmers nicht möglich sein, so regelt das System eine einstellbare Zeitlücke zum vorausfahrenden Fahrzeug. Das System kann das Fahrzeug bei der Längsführung bis in den Stillstand verzögern. Eine Überwachung des Systems ist dauerhaft notwendig, da das System aufgrund der Systemgrenzen nur auf fahrende und anhaltende Verkehrsteilnehmer reagiert. Auf stehende Verkehrsteilnehmer wird nicht reagiert, sodass in diesem Fall der Fahrer eingreifen muss. Da der Fahrer im Rahmen seiner Überwachungsaufgabe nicht mit dem System in Interaktion treten muss, kann das System auch bei nicht stattfindender Überwachung dauerhaft aktiv sein. Aufgrund des Automatisierungsgrads und der oben beschriebenen Systemgrenzen ist dennoch für den sicheren Betrieb des Systems eine dauerhafte Überwachung notwendig.

Das Querführungssystem des Fahrstreifenhalteassistenten (engl. Lane Keeping Assistance System (LKAS)) unterstützt den Fahrer bei der Querführung des Fahrzeugs. Das Fahrzeug wird dabei durch Lenkbewegungen im eigenen Fahrstreifen gehalten (ISO 11270, 2014). In der derzeitigen Ausbaustufe funktioniert das System über einen weiten Geschwindigkeitsbereich, sodass die Einsatzgebiete die Autobahn, die Landstraße und Stadtstraßen sind. Bei der aktuellen Ausprägung des Systems im 7er BMW orientiert sich das System bei Geschwindigkeiten über  $70 \frac{\text{km}}{\text{h}}$  an dem Verlauf der Fahrstreifenmarkierungen, der durch eine Kamera wahrgenommen wird. Bei Geschwindigkeiten unterhalb von  $70 \frac{\text{km}}{\text{h}}$  werden zusätzlich vorausfahrende Fahrzeuge zur Prädiktion des Fahrstreifenverlaufs verwendet (BMW, 2016). Die derzeitige E-Klasse von Mercedes benötigt für die Querführung im Fahrstreifen bis  $130 \frac{\text{km}}{\text{h}}$  keine Fahrstreifenmarkierungen, da sie sich bis zu dieser Geschwindigkeit auch an vorausfahrenden Fahrzeugen oder an Parallelstrukturen zur Straße, wie Leitplanken oder Leitpfosten, orientiert (Kacher, 2016). Bei diesen Systemen stellt der Fahrer die Rückfallebene dar. Der Fahrer muss ebenfalls das System dauerhaft überwachen. Jedoch muss die Überwachung des Systems stetig durch eine Interaktion mit dem System bestätigt werden, damit das System dauerhaft aktiv ist. Die Interaktion wird beispielsweise durch das Berühren des Lenkrads realisiert.

### Automatisierungsgrad 2 - Teilautomatisierung

Bei der Teilautomatisierung übernimmt das System sowohl die Längs- als auch die Querführung für eine gewisse Zeit in spezifischen Situationen. Der Fahrer muss das System bei der Führung des Fahrzeugs dauerhaft überwachen. Die Rückfallebene ist ebenfalls durch den Fahrer abgebildet, der jederzeit bereit sein muss, die vollständige Fahrzeugführung wieder zu übernehmen.

Da bereits die Assistenzsysteme *ACC* und *LKAS* vorgestellt wurden, könnte die Vermutung angestellt werden, dass die Kombination der beiden Systeme bereits ein teilautomatisiertes System darstellen könnte. Dies ist jedoch nicht der Fall, da der Fahrstreifenhalteassistent aufgrund der dauerhaften Interaktion mit dem Fahrer nicht die komplette Querführung des Fahrzeugs übernimmt. Die Querführung ist lediglich über eine kurze Zeitspanne von wenigen Sekunden möglich. Jedoch erfüllen diese wenigen Sekunden nicht den von Gasser u. a. (2012) geforderten „gewissen“ Zeitraum (Reschka, 2015).

Eine beispielhafte Ausprägung eines teilautomatisierten Systems ist der von Bartels (2012) vorgestellte *Autobahn-Assistent*. Der Autobahn-Assistent erfüllt die Teilautomatisierung durch die parallele Ausführung der Längs- und Querführung über einen langen Zeitraum. Dabei handelt es sich, um eine Kombination aus Fahrstreifenhalteassistent und Abstandsregeltempomat mit der Erweiterung, dass der Fahrer nicht dauerhaft mit dem System in Interaktion treten muss, wie es derzeit beim Fahrstreifenhalteassistenten der Fall ist.

Das System hält dabei einen sicheren Abstand zu den übrigen Verkehrsteilnehmern, führt das Fahrzeug in der Mitte des Fahrstreifens und passt die Geschwindigkeit dem vorausfahrenden Verkehrsteilnehmer oder der maximal erlaubten Geschwindigkeit an. Fahrstreifenwechsel oder das Einfädeln in den Verkehrsfluss sind nicht Bestandteil des Systems. Die maximale Geschwindigkeit wird durch digitale Karten oder vorhandene Informationen über wahrgenommene Verkehrszeichen bestimmt (Bartels, 2012). Der Fahrer muss bei diesem Automatisierungsgrad die Fahrfunktion und die Fahrumgebung auf mögliche Systemgrenzen überwachen. Dies bedeutet, dass der Fahrer beispielsweise bei einem falsch erkannten Verkehrszeichen zur Geschwindigkeitsvorgabe oder bei einem Nicht-Erkennen von Objekten in die Längsführung eingreifen muss oder bei einer Fehlinterpretation der Fahrstreifenmarkierungen die Querführung übernehmen muss. Der Fahrer muss jederzeit wieder vollständig die Längs- und Querführung des Fahrzeugs übernehmen können. Eine mögliche Umsetzung des Autobahn-Assistenten als Seriensystem wurde von der Firma Tesla durch die Fahrfunktion *Autopilot*<sup>1</sup> vorgestellt.

### Automatisierungsgrad 3 - Bedingte Automatisierung<sup>2</sup>

Die bedingte Automatisierung, von Gasser u. a. (2012) hochautomatisiert genannt, beschreibt einen weiteren Automatisierungsgrad einer Fahrfunktion. Das System übernimmt gleichzeitig die Längs- und Querführung des Fahrzeugs für einen gewissen Zeitraum in spezifischen Situationen. Der Unterschied zur Teilautomatisierung besteht darin, dass der Fahrer das System nicht dauerhaft überwachen muss. Der Fahrer muss die Führung des Fahrzeugs übernehmen, wenn das System eine Systemgrenze zu überschreiten droht und sich nicht mehr selbst von dieser Systemgrenze entfernen kann. Dabei muss das System die Systemgrenzen selber erkennen.

Damit stellt der Fahrer bei diesem Automatisierungsgrad die Rückfallebene für das System dar. Für die Übernahme muss das System dem Fahrer eine definierte Zeitreserve zur Verfügung stellen, welche für die Übernahme ausreichend ist. Wie groß diese Zeitreserve ist und woran sie sich orientiert, wird von Gasser u. a. (2012) nicht vorgegeben<sup>3</sup>. Bei diesem Automatisierungsgrad ist das System nicht in der Lage, das Fahrzeug in einen sicheren Zustand zu führen, da sich das System nach der definierten Zeitspanne definitiv abschalten wird. Falls der Fahrer zu diesem Zeitpunkt nicht als Rückfallebene zur Verfügung steht, fährt das Fahrzeug unkontrolliert und stellt damit eventuell eine Gefahr für die übrigen Verkehrsteilnehmer und für die Insassen des eigenen Fahrzeugs dar.

---

<sup>1</sup><http://www.sueddeutsche.de/auto/autopilot-im-tesla-model-s-einfach-mal-loslassen-1.2723051>, abgerufen: 02.07.2016

<sup>2</sup>Im Englischen: conditional automation (SAE, 2014)

<sup>3</sup>Welche Zeit für eine Übergabe benötigt wird, steht aktuell noch nicht fest und ist auch stark von der jeweiligen Situation abhängig. Aktuelle Studien gehen davon aus, dass eine Übergabezeit zwischen 5 und 10 Sekunden liegen kann (Damböck, 2013, S.148).

Ein System dieser Stufe ist der von Bartels (2012) vorgestellte *Autobahn-Chauffeur*. Dieser ist eine Weiterentwicklung des *Autobahn-Assistenten*, da bei diesem System der Fahrer das System nicht dauerhaft überwachen muss. Dies bedeutet, dass der Autobahn-Chauffeur seine Systemgrenzen kennt und in der Lage ist, diese zu überwachen. Das System muss in der Lage sein, mögliche Überschreitungen von Systemgrenzen zu erkennen und bei einer drohenden Übertretung mit einer genügend großen Zeitreserve an den Fahrer zu übergeben.

Eine weitere Systemausprägung ist der *Stau-Pilot*<sup>4</sup> von Audi, welcher sich aktuell in der Entwicklung befindet, für den Endkunden jedoch noch nicht verfügbar ist. Dieser übernimmt die Längs- und Querverführung in Stau-Szenarien bei Geschwindigkeiten bis zu  $60 \frac{\text{km}}{\text{h}}$ . Das System übergibt die Fahrzeugführung an den Fahrer, sobald die Geschwindigkeit von  $60 \frac{\text{km}}{\text{h}}$  überschritten wird.

#### Automatisierungsgrad 4 - Hochautomatisierung

Beim hochautomatisierten System, nach der Definition von Gasser u. a. (2012) ein vollautomatisiertes System, wird ebenfalls die Längs- und Querverführung des Fahrzeugs vom System übernommen. Der Fahrer muss das System dabei nicht dauerhaft überwachen. Sollte der Anwendungsfall beziehungsweise die zu beherrschende Fahrsituation verlassen werden, wird die Fahrzeugführung dem Fahrer mit einer genügend großen Zeitreserve übergeben.

Im Unterschied zum bedingt automatisierten System ist das hochautomatisierte System in der Lage, das System in einen sicheren Zustand zurückzuführen. Dies bedeutet im Gegensatz zum bedingt automatisierten System, dass das System alle Systemgrenzen kennen und beherrschen muss. Da das System in jeder Situation in der Lage ist, einen sicheren Zustand herbeizuführen, ist keine Rückfallebene notwendig. Weitere Diskussionen zum sicheren Zustand von automatisierten Fahrfunktionen sind in Reschka und Maurer (2015) zu finden.

Der von Bartels (2012) beschriebene *Autobahn-Pilot* stellt ein, nach der SAE-Definition, hochautomatisiertes System dar. Das System übernimmt die Längs- und Querverführung auf Autobahnen. Der Fahrer muss das System dabei nicht dauerhaft überwachen. Erkennt das System eine Situation, die nicht bewältigt werden kann, versucht das System die Fahrzeugführung an den Fahrer zu übergeben. Sollte die Übergabe nicht möglich sein, wird das Fahrzeug von der Fahrfunktion in einen sicheren Zustand überführt. Dies ist beispielsweise das sichere Anhalten auf dem Seitenstreifen. Dazu muss das System eventuell notwendige Fahrstreifenwechsel durchführen können, um sicher den Seitenstreifen zu erreichen. Eine Ausprägung eines solchen Systems wird von Mirwaldt u. a. (2012) vorgestellt.

#### Automatisierungsgrad 5 - Vollautomatisierung

Den höchsten Automatisierungsgrad nach der SAE Definition stellt die Vollautomatisierung dar. Während die Systeme der vorherigen Automatisierungsgrade nur ausgewählte Anwendungsfälle bewältigen, müssen Systeme vom Grad 5 alle möglichen Situationen bewältigen. Jede Situation muss dabei ohne Überwachung gelöst werden, da gegebenenfalls kein Fahrer im Fahrzeug ist und somit auch nicht als Rückfallebene bereitsteht. Der Betrieb des Fahrzeugs muss daher ohne Fahrer möglich sein. Dieser Grad der Automatisierung wurde bereits von Bartels (2012) angedacht, jedoch von der Forschergruppe nicht weiter verfolgt. Der VDA (2015) nennt diesen Automatisierungsgrad Fahrerlos und ergänzt damit die Terminologie von Gasser u. a. (2012) um den fünften Automatisierungsgrad.

<sup>4</sup>[http://www.focus.de/auto/ratgeber/unterwegs/audi-staupilot-in-shanghai-guck-mal-ohne-haende-so-faehrt-man-im-neuen-audi-a8-durch-jeden-stau\\_id\\_4707009.html](http://www.focus.de/auto/ratgeber/unterwegs/audi-staupilot-in-shanghai-guck-mal-ohne-haende-so-faehrt-man-im-neuen-audi-a8-durch-jeden-stau_id_4707009.html)

Anwendungsfälle für ein vollautomatisiertes Fahrzeug werden von Wachenfeld u. a. (2015) vorgestellt. Diese beschreiben unter anderem das System des *Valet-Parking* als auch das *Fahrzeug auf Abruf* als Anwendungsfälle für vollautomatisierte Fahrfunktionen.

Beim *Valet-Parking* wird das Fahrzeug vom System auf dem nächsten freien Parkplatz geparkt. Eine weitere Anwendung kann das Parken in einem Parkhaus sein, wie es im Projekt *V-Charge*<sup>5</sup> gezeigt wurde. Sollte der Fahrer das Fahrzeug benötigen, wird das Fahrzeug durch das System vom Parkplatz an eine definierte Stelle gefahren und dort dem Fahrer zur Verfügung gestellt.

Beim *Fahrzeug auf Abruf* können sich Personen von einem Punkt zu einem anderen transportieren lassen. Dazu fährt das vollautomatisierte Fahrzeug selbstständig ohne Insassen zum Startpunkt, um die Personen abzuholen. Die Fahrzeuginsassen haben während der Fahrt keine Möglichkeit, die Fahrzeugführung zu übernehmen. Wachenfeld u. a. (2015, S.20) sehen im vollautomatisierten Fahrzeug die Geschäftsmodelle einer „Mischung aus Taxi-Service und Carsharing, autonome Transportfahrzeuge oder Nutzungsmodelle, die über die alleinige Transportaufgabe hinausgehen“.

Tabelle 3.1 zeigt die Automatisierungsgrade nach der Definition der SAE und BASt beziehungsweise VDA. Ebenfalls wird der namentliche Unterschied der Automatisierungsgrade 4 und 5 zwischen den Definitionen der SAE und BASt dargestellt. Die funktionalen Inhalte der Definitionen sind nach der Nummerierung der Automatisierungsgrade identisch.

SAE Level	SAE Name	Ausführung von Quer- und Längsführung	Überwachung der Fahrumgebung	Rückfallebene	Fahrsituationen	BASt / VDA Level
Fahrer überwacht die Fahrumgebung						
0	keine Automatisierung	Fahrer	Fahrer	Fahrer	n.v	„driver only“
1	Assistiert	Fahrer und System	Fahrer	Fahrer	einige	Assistiert
2	Teil-automatisiert	System	Fahrer	Fahrer	einige	Teil-automatisiert
System überwacht die Fahrumgebung						
3	Bedingt automatisiert	System	System	Fahrer	einige	Hoch-automatisiert
4	Hoch-automatisiert	System	System	System	einige	Voll-automatisiert
5	Voll-automatisiert	System	System	System	alle	Fahrerlos

**Tabelle 3.1:** Übersicht der Automatisierungsgrade nach der Bundesanstalt für Straßenwesen (BASt) (Gasser u. a., 2012), dem Verband der Automobilindustrie e.V. (VDA) (VDA, 2015) und Society of Automotive Engineers (SAE) (SAE, 2014).

<sup>5</sup><http://www.v-charge.eu/>, abgerufen: 02.03.2016



## 3.2 Funktionale Systemarchitektur für automatisierte Fahrfunktionen

Von Matthaei (2015) wird eine funktionale Systemarchitektur für automatisiert fahrende Straßenfahrzeuge vorgestellt. Diese basiert auf dem Drei-Ebenen-Modell nach Donges (1999, S.115), welcher Donges (1982) zitiert, mit den Ebenen *Stabilisierung*, *Führung* und *Navigation*. Zusätzlich führt Matthaei (2015, S.39) noch die Detaillierungsgrade mikroskopisch, mesoskopisch und makroskopisch ein, deren Ursprung in der Navigation zu finden ist (Du u. a., 2004). In der Architektur werden die Detaillierungsgrade durch drei Zeilen dargestellt. Besonderes Augenmerk und Forschungsschwerpunkt der Arbeit war die Integration einer wahrnehmungsgestützten Lokalisierung, was die Abhängigkeit von GNSS<sup>6</sup>-Signalen, Korrekturdaten und hochgenauen Karten reduziert (Matthaei, 2015, S.192). Matthaei (2015, Kap.4) präsentiert in seiner Arbeit den Stand der Technik zu funktionalen Systemarchitekturen sowie eine historische Entwicklung der Architektur. Die von Matthaei (2015) entwickelte Architektur wird als Voraussetzung in dieser Arbeit verwendet.

Abbildung 3.1 zeigt den von Matthaei vorgeschlagenen Aufbau der funktionalen Systemarchitektur, erweitert um eine Detaillierung der inneren Modulstrukturen nach Bagschik u. a. (2015c). Im Vergleich zur Architektur von Matthaei (2015) wurden die Spalten *absolute globale Lokalisierung* und *externe Daten* in der Spalte *Lokalisierung & Karten* vereint. Damit unterteilt sich die Architektur in die drei Säulen *Lokalisierung & Karten*, *Umfeld- & Selbstwahrnehmung* und *Planung & Regelung*.

Im Folgenden wird ein Überblick über die funktionale Beschreibung der verschiedenen Module gegeben und die Eingangs- und Ausgangsschnittstellen der Module beschrieben. Diese Beschreibungen werden in Kapitel 9.4 aufgegriffen, um mögliche Testtreiber für den Test von verschiedenen Modulen der funktionalen Systemarchitektur zu diskutieren.

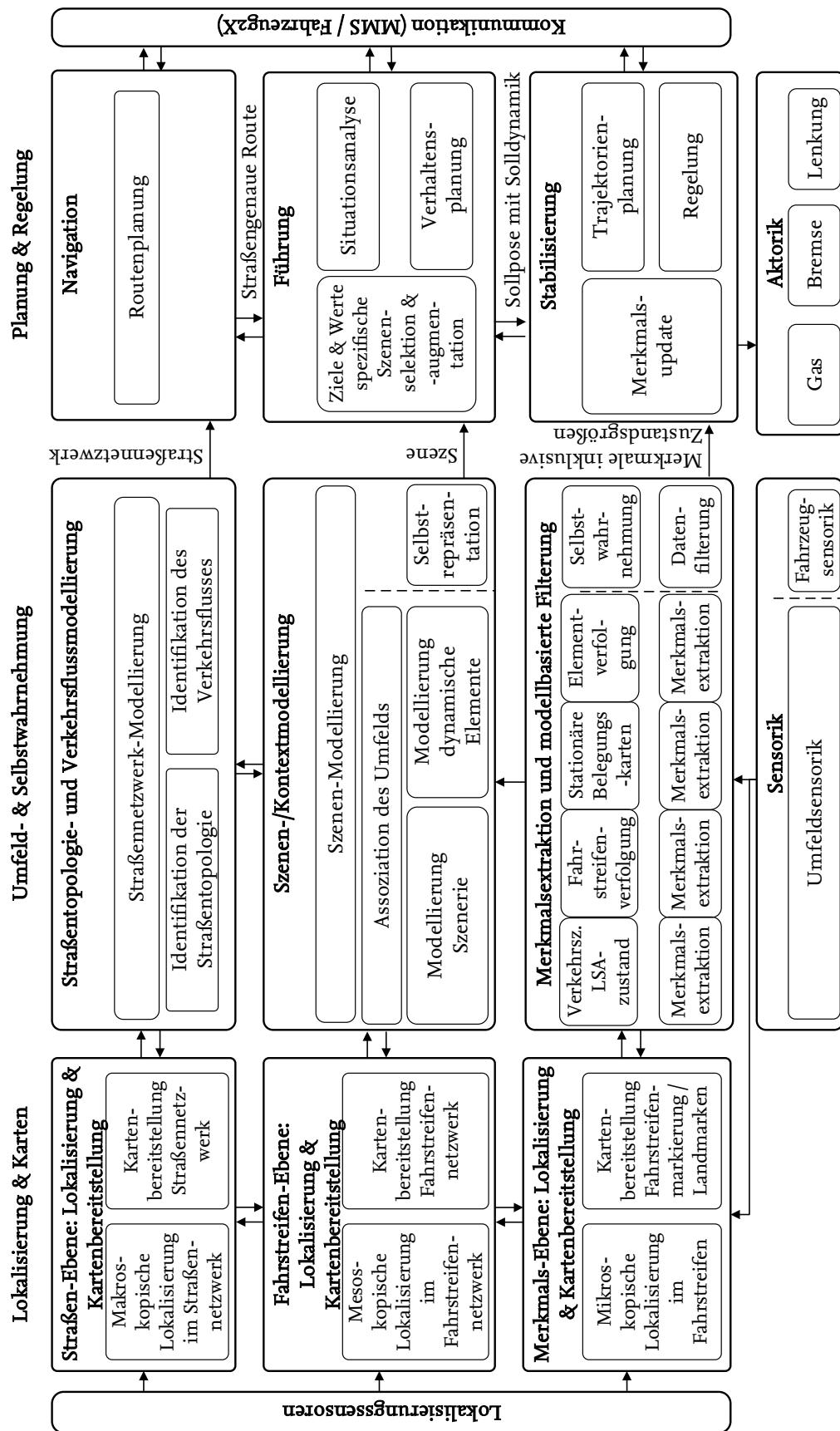
### 3.2.1 Lokalisierung & Karten

Die Lokalisierung findet auf einer makroskopischen, mesoskopischen und mikroskopischen Ebene statt. Diese Ebenen werden ebenfalls für die Unterscheidung von verschiedenen Abstraktionsebenen von Karten verwendet.

Eine Lokalisierung auf makroskopischer Ebene ist straßengenau und hat eine Genauigkeit von ca. 10 - 20 Meter. Die Lokalisierung wird mit handelsüblichen Lokalisierungssensoren, wie GNSS-Empfängern, erreicht. Mesoskopische Lokalisierungen haben eine Genauigkeit von ca. 1,5 Meter und ermöglichen eine Lokalisierung im Fahrstreifen. Diese Genauigkeit der Lokalisierung ist mit handelsüblichen GNSS-Empfängern nicht möglich. Für eine mesoskopische Lokalisierung werden Verfahren, wie ein differentielles globales Positionierungssystem (DGPS) oder die Fusion von GNSS und Inertialsensorik, benötigt. Mikroskopische Lokalisierung findet nach Matthaei (2015, S.39f) mit einer quasi-kontinuierlichen Genauigkeit statt. Um Positionierungen mit dieser Genauigkeit zu erreichen, sind Verfahren, wie DGPS mit einer Fusion mit Inertialsensorik, notwendig. Matthaei (2015, S.134) weist darauf hin, dass diese Verfahren in städtischen Szenarien, durch die Abschattung der GNSS-Signale durch Häuser oder ähnliche Gebäude, nicht immer eine mikroskopische Position bestimmen können. Daher sind Karten und eine Umfeldwahrnehmung zusätzlich für eine Lokalisierung erforderlich.

---

<sup>6</sup>GNSS: Globales Navigationssatellitensystem



**Abbildung 3.1:** Funktionale Systemarchitektur für automatisierte Fahrfunktionen nach Matthaei (2015) und detailliert nach Bagschik u. a. (2015c). Im Vergleich zur Architektur von Matthaei (2015) wurden die Spalten absolute globale Lokalisierung und externe Daten in der Spalte Lokalisierung & Karten vereint. Abkürzung: LSA: Lichtsignalanlage

Die Karten sind neben der Lokalisierung der zweite Bestandteil dieser Spalte der Architektur. Die Karten werden benötigt, um mittels Lokalisierung eine kartenrelative Pose zu bestimmen. Für die Karte werden ebenfalls verschiedene Abstraktionsstufen unterschieden. Bei einer straßengenauen makroskopischen Karte werden die Straßen ohne Informationen über die Anzahl der Fahrstreifen und deren Funktion in Form eines gerichteten Graphen dargestellt. Eine mesoskopische Karte fügt zusätzlich die Fahrstreifenmitten zur Karte hinzu. Bei einer mikroskopischen Karte werden weitere Informationen, zum Beispiel über die Markierungen der Fahrstreifen oder die Fahrstreifenbreiten, ergänzt. Für die Lokalisierung innerhalb der Karten werden die Eingangsdaten der Lokalisierungssensoren und wahrgenommene Merkmale, wie Landmarken, verwendet. Die Pose des Fahrzeugs in einer definierten Genauigkeit ist die Ausgabe dieser Spalte (Matthaei, 2015, S.42ff).

### 3.2.2 Umfeldwahrnehmung und Selbstwahrnehmung

Die zweite Spalte der funktionalen Systemarchitektur stellt die Umfeld- und Selbstwahrnehmung dar. Diese lässt sich in die Ebenen der Sensorik, Merkmalsextraktion und modellbasierte Filterung, der Szenen-/Kontextmodellierung sowie der Straßentopologie- und Verkehrsflussmodellierung unterteilen. Nach Matthaei (2015, S.48) stellt die Wahrnehmung das zentrale Element der Architektur dar. Es werden die Informationen für die Planung und Regelung bereitgestellt, die für die Erfüllung der Mission benötigt werden, und die Informationen, die zur kartenrelativen Lokalisierung genutzt werden, extrahiert. Im Folgenden werden die einzelnen Blöcke von unten nach oben, entsprechend dem Fluss der Erstellung und Interpretation der Informationen, skizziert.

#### Sensorik

Die Sensorik unterteilt sich in Umfeldsensorik und Fahrzeugsensorik. Zur Fahrzeugsensorik zählen beispielsweise Odometer, Gyroskop, Füllstandssensor des Kraftstofftanks, Lenkwinkelsensor, Lagesensor, Beschleunigungssensor, etc. Dies sind dementsprechend Sensoren, die den inneren Zustand des Fahrzeugs erfassen. Die Umfeldsensorik nimmt Informationen über Elemente wahr, die sich außerhalb des Fahrzeugs befinden. Sensoren dieser Art sind Kameras, Lidar, Radar, Ultraschall, Regensensor, etc. Die Ausgangsdaten der Sensoren stellen Informationen über das Fahrzeug und das Umfeld dar. Diese Informationen werden im nachfolgenden Modul der Merkmalsextraktion und modellbasierten Filterung weiterverarbeitet.

#### Merkmalsextraktion und modellbasierte Filterung

In der Merkmalsextraktion und modellbasierten Filterung werden die Ausgangsdaten der Lokalisierung in Form einer Pose und Eigenbewegungsschätzung verarbeitet. Weiterhin werden die Ausgangsdaten der Sensorik in den verschiedenen Modulen zur Extraktion von weiteren Informationen verwendet. Die Module dienen derzeit zur Verfolgung von dynamischen Elementen und Fahrstreifen, zur Erstellung von stationären Belegungskarten, zur Schätzung des Zustands von Lichtsignalanlagen sowie der Erkennung von Verkehrszeichen. Die Architektur bietet eine Erweiterung dieser Liste. Zunächst findet stets eine Merkmalsextraktion statt, um relevante Merkmale aus den Eingangsdaten zu extrahieren. Darauf aufbauend werden nach Matthaei (2015, S.50) verschiedene Filter und Modellannahmen in den Modulen verwendet, um Informationen aus den Eingangsdaten zu generieren.

Daneben findet eine Datenfilterung der Eingangsdaten für eine Selbstwahrnehmung des Fahrzeugs statt. Reschka (2016a, S.4) beschreibt die Selbstwahrnehmung als „die Erfassung



und Interpretation von Messwerten durch die Sensorik eines Systems“. Das Thema der Selbstwahrnehmung wird in dieser Arbeit nicht weiter betrachtet. Weitere Informationen sind in Bergmiller (2014), Reschka u. a. (2012, 2015a) und Reschka (2016a) zu finden.

Die Ausgänge der Merkmalsextraktion und der modellbasierten Filterung sind Informationen über dynamische Elemente, wie die Abmessungen, relative Position und Geschwindigkeit, sowie die befahrbaren Bereiche, welche derzeit in der Architektur durch Belegungskarten beschrieben werden. Weiterhin werden Informationen über die Fahrstreifen in Form von Klothoiden oder Linienzügen sowie Informationen über die Fahrstreifenmarkierungen, wie die Farbe oder Breite, und die relative Lage des Fahrzeugs zum Fahrstreifen weitergegeben. Daneben werden Informationen über erkannte Verkehrszeichen und der Zustand sowie die Position von Lichtsignalanlagen an die folgenden Module übertragen.

### **Szenen-/Kontextmodellierung**

In der Szenen- beziehungsweise Kontextmodellierung werden die Eingangsinformationen der Lokalisierung, Merkmalsextraktion und der modellbasierten Filterung verwendet, um Szenen, nach der Definition in Kapitel 2.3, zu modellieren. Dabei werden die Szenerie und die dynamischen Elemente modelliert, wodurch stationäre und dynamische Elemente voneinander getrennt werden. Die Informationen werden für eine Assoziation des Umfeldes genutzt, um Szenen zu erstellen. Dabei werden dynamische Elemente den Fahrstreifen zugeordnet und mit Hilfe von mikroskopischen Karten die Positionen der Fahrzeuge innerhalb der Fahrstreifen bestimmt. Weiterhin werden, mit Hilfe von digitalen Karten, Kreuzungstopologien und kartografierte Fahrstreifennetze mit erfassten Fahrstreifen verglichen. Dabei werden Lichtsignalanlagen den Fahrstreifen zugeordnet und Umweltbedingungen der Szenerie hinzugefügt. Die Selbstrepräsentation wird aus der Selbstwahrnehmung generiert und der Szene hinzugefügt. Ausgangsinformationen sind einzelne Szenen mit einem Zeitstempel. Die Szenen werden den Modulen der Straßentopologie- und Verkehrsflussmodellierung sowie dem Modul der Führung übergeben.

### **Straßentopologie- und Verkehrsflussmodellierung**

Das Modul der Straßentopologie- und Verkehrsflussmodellierung erhält seine Eingangsinformationen von der Szenen-/Kontextmodellierung, der makroskopischen Lokalisierung und der Kartenbereitstellung. Ziel ist die Identifikation der Straßentopologie und des Verkehrsflusses. Durch die Eingangsdaten in Form von Szenen wird der Verkehrsfluss in der Umgebung des Fahrzeugs geschätzt. Die Lokalisierung wird verwendet, um die Position im Straßennetzwerk zu bestimmen. Durch zusätzliche Informationsquellen, wie Staumeldungen, wird die makroskopische Karte mit weiteren Informationen erweitert. Das Modul bietet als Ausgangsschnittstelle ein Straßennetzwerk, welches Informationen über den aktuellen Verkehrsfluss und die Verkehrsdichte enthält.

### **3.2.3 Planung und Regelung**

Die dritte Säule in der funktionalen Systemarchitektur stellt die Planung und Regelung dar. Diese umfasst die Module der Navigation, Führung, Stabilisierung und Aktorik. Im Folgenden werden die einzelnen Module beschrieben. Die Reihenfolge der Beschreibung orientiert sich am Aggregationslevel der Informationen, die von den Modulen verarbeitet und ausgegeben werden.

### **Navigation**

Die Navigation erhält als Eingangsdaten die Informationen über das aktuelle Straßennetz inklusive der Verkehrsdichte und die Mission des Fahrzeugs, wie das Erreichen einer Zieladresse. Basierend auf diesen Daten berechnet die Routenplanung eine nach definierten Kriterien optimale Route. Diese Funktionalität ist bereits in Serienfahrzeugen durch die Wahl der schnellsten, kürzesten oder ökonomischsten Route verfügbar. Aufgerufen wird die Routenplanung nicht zyklisch, sondern ereignisbasiert. Ein mögliches Ereignis ist beispielsweise eine Änderung der Auslastung des Straßennetzwerks. Weiterhin ist es möglich, dass das Modul der Führung der geplanten Route nicht folgen kann. Dies wird der Routenplanung vom Führungsmodul mitgeteilt und führt zu einer Neuberechnung der Route.

### **Führung**

Das Modul der Führung erhält die aktuelle Szene und eine straßengenaue Route, um die vorgegebene Mission zu erfüllen. Innerhalb des Moduls findet eine ziel- und wertspezifische Szenenselektion und -augmentation statt, wodurch aus der Szene eine Situation erstellt wird. Diese wird in der Situationsanalyse bewertet. Von der Verhaltensplanung wird darauf aufbauend eine Liste von Manövern erstellt, die zur Erfüllung der Mission durchgeführt werden müssen. Dies kann zum Beispiel die Planung eines Fahrstreifenwechsels sein, um an einer Kreuzung abzubiegen. Im Führungsmodul werden für die Bewertung der Situation und Planung der Manöver ebenfalls Informationen aus der Stabilisierung benötigt. Dies kann unter anderem die aktuell geplante Trajektorie sein. Daher existiert ein Rückkanal zwischen der Stabilisierung und dem Modul der Führung. Die Ausgangsschnittstelle zum Modul der Stabilisierung ist eine zu erreichende Sollpose mit Solldynamik. Die Solldynamik beschreibt die Geschwindigkeit und Beschleunigung in der Sollpose.

### **Stabilisierung**

Die Stabilisierung erhält die Sollpose mit Solldynamik und die extrahierten Merkmale inklusive der Zustandsgrößen aus dem Modul der Merkmalsextraktion und modellbasierten Filterung. Mit Hilfe der Eingangsdaten wird mittels eines Algorithmus zur Trajektorienplanung eine Trajektorie zur Erreichung der Sollpose berechnet. Diese wird der Regelung übergeben, welche Stellgrößen für die Aktorik berechnet. Durch die Merkmalsupdates können die Merkmale mit einer hohen Frequenz der Stabilisierung übergeben werden und somit die Regelung der Trajektorie realisiert werden. Die Ausgangsdaten der Stabilisierung sind die Stellgrößen der Aktorik. Dies kann ein Beschleunigungswunsch oder eine Gaspedalstellung beziehungsweise Bremspedalstellung oder ein Lenkwinkel beziehungsweise Lenkmoment sein, je nachdem, welche Schnittstellen zum Fahrzeug verfügbar sind.

### **Aktorik**

Die Aktorik stellt die Hardware zur Umsetzung der Stellgrößen dar. Typische Aktorikkomponenten im Fahrzeug sind Lenkung, Motor, Getriebe oder Bremsen, die die Stellgrößen der Stabilisierung umsetzen.

#### **3.2.4 Kommunikation**

Die Kommunikation beschreibt den bidirektionalen Austausch von Informationen zwischen der automatisierten Fahrfunktion und den Insassen des Fahrzeugs sowie den übrigen Verkehrsteilnehmern. Die Kommunikation kann auf optische, haptische und/oder akustische Weise realisiert werden.

### 3.3 Exemplarische Fahrfunktion zur Veranschaulichung des Testkonzepts

Dieses Unterkapitel stellt eine automatisierte Fahrfunktion vor, anhand derer die exemplarische Anwendung des Testbaukastens im Verlauf der vorliegenden Arbeit gezeigt wird. Dabei hat die Anwendung keinen Anspruch auf Vollständigkeit, da eine vollständige Betrachtung einer Fahrfunktion im Rahmen einer Dissertation nicht durchführbar ist. Das Beispiel zeigt lediglich die grundsätzliche, theoretische Vorgehensweise des Testbaukastens. Eine praktische Anwendung des Testbaukastens wird in Kapitel 12 der vorliegenden Arbeit anhand des Engstellenassistenten vorgestellt. Die Anwendung des Engstellenassistenten ist als direktes Beispiel während der Kapitel zu umfangreich, wodurch an dieser Stelle die exemplarische Fahrfunktion eingeführt wird.

Die ausgewählte automatisierte Fahrfunktion ist eine Teilfunktion des *Autobahn-Piloten*, der von Bartels (2012) vorgestellt wird und ein System mit Automatisierungsgrad 4 darstellt. Der *Autobahn-Pilot* übernimmt dauerhaft die parallele Ausführung der Längs- und Querführung des Fahrzeugs in der Verkehrsdomäne Autobahn. Eine dauerhafte Überwachung des Systems durch den Fahrer ist nicht notwendig. Für die vereinfachte Anwendung des Testbaukastens wird die Querführungsfunktion aus dem Gesamtsystem herausgegriffen. Von der Querführungsfunktion wird die Teilfunktion Fahrstreifenhalten ohne Fahrstreifenwechsel betrachtet. Diese Teilfunktion ist in den von Bartels (2012) beschriebenen Fahrfunktionen *Autobahn-Assistent* mit Automatisierungsgrad 2 und *Autobahn-Chauffeur* mit Automatisierungsgrad 3 vorhanden. Für eine Fahrfunktion mit Automatisierungsgrad 5 wird die beschriebene Teilfunktion ebenfalls benötigt. Damit kann, begrenzt auf die gewählte Teilfunktion, die theoretische Anwendbarkeit des Testbaukastens auf Fahrfunktionen mit Automatisierungsgrad 2 - 5 gezeigt werden. Bei den Automatisierungsgraden 2 und 3 stellt der Fahrer die Rückfallebene des Systems dar. Weiterhin hat der Fahrer bei den Automatisierungsgraden 2 - 4 die Möglichkeit, die Fahrzeugführung zu übernehmen. Die Herausforderungen, die sich daraus ergeben, werden ebenfalls bei der Beschreibung der Anwendung des Testbaukastens diskutiert.

Die Architektur des Systems orientiert sich an der funktionalen Systemarchitektur aus Kapitel 3.2. Als Umfeldsensorik wird eine Kamera verwendet. Auf Basis der aufgenommenen Bilder wird eine Merkmalsextraktion und Fahrstreifenverfolgung durchgeführt. Die identifizierten Fahrstreifen werden der Trajektorienplanung übergeben, welche eine Trajektorie berechnet, um dem eigenen Fahrstreifen zu folgen. Durch die Regelung werden Sollgrößen berechnet, um der Trajektorie zu folgen. Diese Sollgrößen werden an die Lenkung übergeben. Im Rahmen der Tests soll die Querführungsfunktion auf Autobahnen ohne weitere bewegliche Verkehrsteilnehmer getestet werden. Dabei soll ebenfalls die Übergabe an den Fahrer getestet werden. Im Rahmen der Anwendung des Testbaukastens sollen X-in-the-Loop-Verfahren und Testtreiber, wie sie in Kapitel 9.4 vorgestellt werden, für die Testdurchführung identifiziert werden.

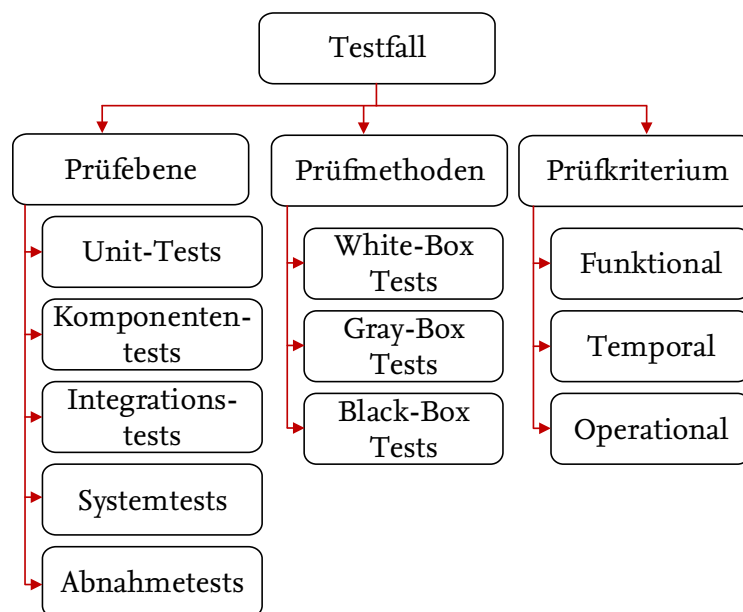


## 4 Methoden des Softwaretests und der Verkehrssimulation

Dieses Kapitel beschreibt Methoden des Softwaretests anhand des V-Modells sowie verschiedene Abstraktionsebenen der Verkehrssimulation. Damit werden die Grundlagen für die Definition von Anforderungen des modularen virtuellen Testbaukastens an Softwaretestmethoden und Methoden der Verkehrssimulation vorgestellt. Zusätzlich wird eine Klassifikationsmethode für X-in-the-Loop-Verfahren präsentiert, die im Testbaukasten für die Zuordnung von Testfällen auf X-in-the-Loop-Verfahren verwendet wird.

### 4.1 Softwaretestmethoden

Ein Testfall wird nach Hoffmann (2008, S. 158) anhand der Merkmale Prüfebene, Prüfmethodik und Prüfkriterium klassifiziert. Abbildung 4.1 stellt dies dar. Die Merkmale werden im Folgenden verwendet, um verschiedene Softwaretestmethoden vorzustellen.



**Abbildung 4.1:** Merkmalsraum der Testklassifikation in Anlehnung an Hoffmann (2008) erweitert um den Komponententest

Unit-Tests, Komponententests, Integrationstests, Systemtests sowie Abnahmetests stellen Tests auf verschiedenen Prüfebenen dar. Diese Ebenen werden im V-Modell nach VDI 2206 (2004) als Entwicklungsebenen vorgesehen. Das V-Modell wird in dieser Arbeit als etablierter Prozess zur Entwicklung und zum Test von automatisierten Fahrfunktionen verwendet, da das V-Modell ebenfalls in existierenden Standards wie der ISO 26262 (2011, Teil 3,6) eingesetzt wird. Daher werden in der vorliegenden Arbeit keine weiteren Entwicklungsmethoden betrachtet. Die Prüfmethoden lassen sich in Black-Box, White-Box und Gray-Box Methoden

unterscheiden. Das Prüfkriterium ist nach Hoffmann (2008, S.170ff) funktional, operational oder temporal. Funktionale Prüfkriterien testen die Funktionsweise eines Testobjekts. Diese können durch Funktionstests, Trivialtests, Crashtests oder Zufallstests nachgewiesen werden. Beim Funktionstests wird überprüft, ob das Testobjekt bei vorgegebenen Eingangsgrößen die Werte der Ausgangsgrößen korrekt berechnet (Hoffmann, 2008, S.170). Bei operationalen Tests wird das Testobjekt im Einsatz getestet, wobei mögliche Tests Installationstests, Ergonomietests oder Sicherheitstests sind. Temporale Tests prüfen die Software über einen längeren Zeitraum durch Komplexitätstests, Laufzeittests, Lasttests oder Stresstests. In den folgenden Abschnitten werden die verschiedenen Prüfmethoden und Prüfebenen vorgestellt. Auf die Prüfkriterien wird im Rahmen der vorliegenden Arbeit nicht weiter eingegangen, da der modulare virtuelle Testbaukasten lediglich funktionale Tests durchführt. Weitere Informationen zu temporalen und operationalen Prüfkriterien können in Hoffmann (2008, Kap. 4.2) gefunden werden.

#### 4.1.1 Prüfmethoden

Die Prüfmethoden lassen sich in statische und dynamische Methoden unterteilen. Die statischen Prüfmethoden werden angewendet, bevor das Testobjekt in Betrieb genommen wird. Bei den dynamischen Prüfmethoden ist das Testobjekt während der Tests in Betrieb. Zu der Gruppe der statischen Prüfmethoden gehören unter anderem die analysierenden und verifizierenden Methoden. Ein mögliches Verfahren ist die formale Verifikation, die nach Beyer (2002, S.5) überprüft, ob „das Modell, die in der formalen Spezifikation fixierten Eigenschaften hat (Korrektheitsbeweis); falls ja, dann gilt das Modell als korrekt bezüglich der Spezifikation“. Da der Testbaukasten das Testobjekt in verschiedenen Szenarien testet und die Ausgangsdaten des Testobjekts für die Berechnung neuer Eingangsdaten benötigt werden, muss das Testobjekt während der Tests in Betrieb sein. Daher werden die statischen Prüfmethoden in der vorliegenden Arbeit nicht weiter betrachtet und es werden lediglich die dynamischen Prüfmethoden für den Testbaukasten bewertet. Die dynamischen Prüfmethoden lassen sich in White-Box-, Black-Box- und Gray-Box-Methoden unterteilen. Dynamische Verfahren haben einen Stichprobencharakter, da nicht alle möglichen Situationen getestet werden können (Liggesmeyer, 2009, S.39). Daher treffen diese Testmethoden nur eine unvollständige Aussage über die Korrektheit des Testobjekts (Liggesmeyer, 2009, S.39). Bei der Verwendung von dynamischen Tests ist immer eine verbleibende Fehlerwahrscheinlichkeit vorhanden. Dijkstra (1972, S.864) hat dies im folgenden Zitat zusammengefasst:

„Durch Testen kann man stets nur die Anwesenheit, nie aber die Abwesenheit von Fehlern beweisen.“

#### White-Box-Tests

White-Box-Tests nutzen detaillierte Modelle über das Testobjekt sowie Informationen über den inneren Aufbau beziehungsweise die Struktur des Testobjekts, um Testfälle zu konstruieren. Dabei werden nach Hoffmann (2008, S.201) unter anderem kontrollflussorientierte oder datenflussorientierte Methoden verwendet. Diese Methoden testen jede Anweisung, jeden Entscheidungsweig oder jede Bedingung auf ihre Korrektheit. Die Eingangsdaten für die Tests werden aus der inneren Struktur und den detaillierten Modellen des Testobjekts abgeleitet. Ein Vorteil der White-Box Tests liegt in den vorhandenen Metriken zur Testabdeckung beziehungsweise Testvollständigkeit. Durch die Tests wird theoretisch nachgewiesen,

dass alle Anweisungen und Bedingungen fehlerfrei ausgeführt werden (Hoffmann, 2008, S.201). Praktisch ist dies aber aufgrund von Zeit- und Kostenlimitationen nicht möglich (Sommerville, 2006, S.539).

### Black-Box-Tests

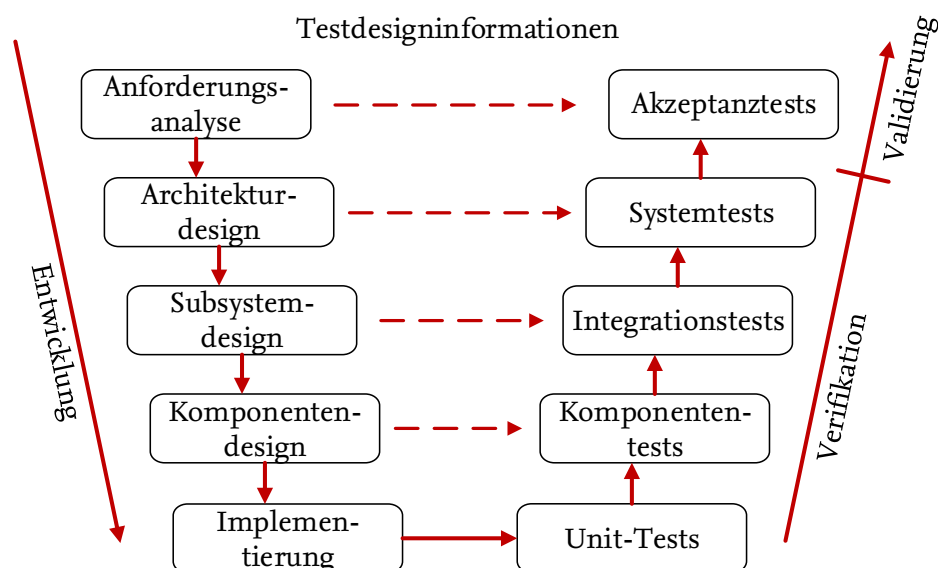
Für Black-Box-Tests werden die Ein- und Ausgangsdaten des Testobjekts für die Konstruktion von Testfällen verwendet. Die Testfälle werden auf Basis der erwarteten Ausgangsdaten und des erwarteten Verhaltens bewertet. Der Quellcode oder der Aufbau der inneren Struktur sind nicht bekannt und steht dementsprechend nicht für die Konstruktion der Testfälle zur Verfügung. Standardmethoden für Black-Box Tests sind Äquivalenzklassentests (Spillner und Linz, 2012, S.114ff), Grenzwertbetrachtungen (Arnicane, 2009) und kombinatorische Tests (Grindal u. a., 2005; Nie und Leung, 2011), die funktionsorientierte Kriterien überprüfen. Diese Methoden werden im modularen virtuellen Testbaukasten verwendet.

### Gray-Box-Tests

Die Gray-Box-Tests verbinden die Vorteile der White-Box und Black-Box-Tests. Die Testfälle werden nach den Methoden des Black-Box-Testens erstellt. Jedoch werden für die Erstellung der Testfälle Informationen über die innere Struktur des Testobjekts benötigt, wodurch die Voraussetzungen für die Gray-Box-Tests zum Teil den Voraussetzungen von White-Box-Tests entsprechen. Detaillierte Modelle des Testobjekts sind jedoch nicht unbedingt notwendig. Jeder Testfall, der vom Entwickler des Moduls erstellt wird, stellt somit mindestens einen Gray-Box-Test dar (Hoffmann, 2008, S.174).

## 4.1.2 Prüfebenen nach dem V-Modell

Der in dieser Arbeit verwendete Entwicklungsprozess von Software ist das V-Modell, welches unter anderen von Rook (1986), Ammann und Offutt (2008, S.6) beziehungsweise der ISO 26262 (2011)[Teil 6] vorgestellt wird. Abbildung 4.2 zeigt das Entwicklungsmodell nach Rook (1986) mit angepasster Terminologie der ISO 26262 (2011, Teil 1).



**Abbildung 4.2:** V-Modell nach Rook (1986), angepasst an die Terminologie der ISO 26262 (2011, Teil 1)



Im linken Teil des V-Modells sind die Schritte zur Entwicklung des Systems abgebildet. Zu Beginn wird eine Anforderungsanalyse durchgeführt. Darauf aufbauend wird das Design der Architektur, der Subsysteme und der Komponenten festgelegt. Auf den jeweiligen Stufen werden Informationen für die Spezifikation von Testfällen gesammelt, um das System auf der rechten Seite des Modells zu verifizieren und zu validieren. Der linke Teil des V-Modells wird in der vorliegenden Arbeit nicht weiter analysiert, da der Testbaukasten lediglich die rechte Seite des V-Modells betrachtet. Weitere Informationen zu Methoden der linken Seite des Modells sind unter anderem in Reschka (2016a, Kap. 4) zu finden, der eine Anforderungsanalyse für ein vollautomatisiertes Fahrzeug in seiner Arbeit betrachtet.

Zur Verifikation werden Unit-Tests, Komponententests, Integrationstests und Systemtests angewendet. Für die Validierung werden Abnahmetests durchgeführt. Die Tests können mit oder ohne Rückführung (engl. open-loop und closed-loop) der Ausgaben des Testobjekts durchgeführt werden. Bei einem Test mit Rückführung werden die Ausgaben des Testobjekts über einen Testtreiber, zum Beispiel Simulationsmodelle, verarbeitet und daraus neue Eingangsdaten für das Testobjekt berechnet. Somit haben die Ausgaben des Testobjekts einen direkten Einfluss auf zukünftige Eingangsdaten. Bei einem Test ohne Rückführung werden die Ergebnisse des Testobjekts nicht für die Berechnung weiterer Eingangsdaten beachtet. Das Testobjekt hat somit nicht die Möglichkeit, die zukünftigen Eingangsdaten zu beeinflussen. Vergleichbar sind diese beiden Testarten mit offenen und geschlossenen Regelkreisen. Im Folgenden werden die verschiedenen Prüfebenen beschrieben.

### **Software Unit-Tests**

Eine Software Unit bezeichnet nach der ISO 26262 (2011, Teil 1, S.16) eine atomare Softwareeinheit der Softwarearchitektur, die eine einzelne Methode einer Softwareklasse sein kann. Für Tests solcher Units werden Methoden des White-Box Testens verwendet, da die Units direkt den Quellcode darstellen und somit auf dieser Basis getestet werden können (Ammann und Offutt, 2008, S.6). Mit den Methoden des White-Box Testens können, wie oben beschrieben, Pfad- und Anweisungsabdeckungen getestet und damit die Implementierung der Unit bewertet werden (Ammann und Offutt, 2008, S.7). Ein Werkzeug, welches auf dieser Ebene verwendet wird, ist die kontinuierliche Integration und der kontinuierliche Test<sup>1</sup> (engl. continuous integration and testing), welche automatisiert entwicklungsbegleitende Tests durchführen.

### **Komponententests**

Eine Komponente wird nach der ISO 26262 (2011, Teil 1, S.3) allgemein als ein Element bezeichnet, welches kein Systemelement darstellt. Eine Softwarekomponente wird nach der ISO 26262 (2011, Teil 1, S.16) speziell als eine Zusammenstellung von mehreren Software-Units beschrieben. In der Literatur werden Unit-Tests und Komponententests oft auf einer Stufe gesehen (Hoffmann, 2008, S.158). In dieser Arbeit wird explizit eine Trennung zwischen Units und Komponenten vorgenommen. Während Units als Softwareteilkomponenten gesehen werden, werden Komponenten in dieser Arbeit, wie in der ISO 26262 (2011, Teil 1, S.16) und Ammann und Offutt (2008, S.6) beschrieben, als abgeschlossene Elemente gesehen, die aus einer Vielzahl von einzelnen Units bestehen können. Die Komponenten können beispielsweise einzelne Module der funktionalen Systemarchitektur sein, welche in Kapitel 3.2 vorgestellt wurde. Durch diese Unterscheidung ist es möglich, den Testprozess von einzelnen

---

<sup>1</sup>Beispielsweise durch die Software Jenkins: <https://jenkins-ci.org/>



Softwareteilkomponenten, die mit White-Box Methoden testbar sind, von Komponententests zu trennen, bei denen die innere Struktur des Testobjekts nicht bekannt ist und somit mit Black-Box Methoden getestet werden müssen. Die Komponententests werden, wie Unit-Tests, entwicklungsbegleitend durchgeführt.

### **Integrationstests**

Auf der nächsten Prüfebene werden Integrationstests durchgeführt, die die Schnittstellen und die Interaktionen zwischen verschiedenen Komponenten testen. Das Zusammenfügen von Units zu Komponenten stellt ebenfalls einen Integrationstest dar. Die Testmethoden werden unter anderem in die Methoden der Big-Bang-Integration und strukturorientierten Integration unterteilt (Hoffmann, 2008, S.163). Bei einer Big-Bang-Integration werden alle Komponenten erst nach der vollständigen Implementierung im Verbund getestet. Nachteil ist ein erhöhter Testaufwand, da die Lokalisierung der Ursache von Fehlern mit erhöhtem Aufwand verbunden ist (Sommerville, 2006, S.33). Neben diesem Nachteil hat die Methode den Vorteil, dass alle Komponenten bereits vollständig implementiert sind und keine Testtreiber für die Erstellung von Testdaten der vor- oder nachgelagerten Komponenten benötigt werden. Bei der strukturorientierten Integration werden die Komponenten nach der Systemstruktur integriert. Je nachdem, welche Strategie verwendet wird, werden verschiedene Testtreiber für die vor- oder nachgelagerten Komponenten benötigt. Die Integrationstests werden ebenfalls entwicklungsbegleitend durchgeführt.

### **Systemtests**

Nach den Integrationstests werden in der nächsten Prüfebene Systemtests durchgeführt. Bei diesen Tests werden ausschließlich Methoden des Black-Box Testens eingesetzt (Liggesmeyer, 2009, S.435). Für die Tests werden keine Informationen über die innere Struktur des Systems beachtet, da ausschließlich das funktionale Verhalten des Gesamtsystems getestet wird. Das System wird lediglich auf Basis von Eingangs- und Ausgangsdaten sowie auf der Basis von vorhandenen Spezifikationen getestet und bewertet. Das primäre Ziel von Systemtests ist die Aufdeckung von unvorhergesehenen Fehlern bei der vollständigen Integration des Systems und von unklaren Anforderungen, die nur vage formuliert wurden (Hoffmann, 2008, S.167).

### **Abnahmetest**

Der Abnahmetest verfolgt ähnliche Ziele wie der Systemtest. Jedoch unterscheidet sich der Abnahmetest vom Systemtest in einem Aspekt. Während bei den Prüfebenen von Unit-Tests bis Systemtests eine Verifikation (siehe Definition in Kapitel 2.2) der Units, Komponenten und des Systems stattfindet, wird beim Abnahmetest eine Validierung (siehe Definition in Kapitel 2.2) durchgeführt. Damit liegt beim Abnahmetest die Verantwortung der Abnahme zum ersten Mal in der Hand des Auftraggebers und nicht in der des Systementwicklers oder Auftragnehmers (Hoffmann, 2008, S.168). Dadurch stellt der Abnahmetest auch einen Freigabetest des Systems dar. Für eine automatisierte Fahrfunktion könnte beispielsweise die Freigabe für den Straßenverkehr durch die Abnahme beim technischen Überwachungsverein (TÜV) durchgeführt werden. Neben der funktionalen Bedeutung hat der Akzeptanztest eine juristische Relevanz, da das Ergebnis der Tests direkt den Grad der Vertragserfüllung zwischen Auftraggeber und Auftragnehmer festlegt (Hoffmann, 2008, S.168).

## 4.2 Verkehrssimulationen

Dieses Kapitel stellt verschiedene Abstraktionsebenen der Verkehrssimulation vor, um diese in Kapitel 9.1.2 für die Bewertung der Verwendbarkeit im modularen virtuellen Testbaukasten zu nutzen. Verkehrssimulationen für automatisierte Fahrfunktionen lassen sich nach Weillkes (2000, S.60) in die Ebenen makroskopisch, mikroskopisch und submikroskopisch unterteilen. Von Rose (2003, S.3) wird die mesoskopische Verkehrssimulation als Ebene zwischen der makroskopischen und mikroskopischen Simulation gesehen. Eine ähnliche Einteilung wird von Matthaei (2015) verwendet, um in der vorgestellten Systemarchitektur die funktionalen Abstraktionsebenen zu trennen. Dabei orientiert er sich an Du u. a. (2004), die die Ebenen für eine Einordnung von Navigationsaufgaben für automatisierte Fahrzeuge verwenden. Im Folgenden werden die Ebenen beschrieben und ein Vergleich zu Du u. a. (2004) hergestellt, um vorhandene Unterschiede zu identifizieren.

### 4.2.1 Makroskopische Verkehrssimulationen

Nach Kemper (2006, S.53) liegt der Fokus der makroskopischen Verkehrssimulation auf der Beobachtung der kollektiven Gesamtdynamik des Verkehrsflusses. Makroskopische Verkehrssimulationen modellieren den Verkehr auf Basis von Fahrzeugkollektiven, bei denen die Fahrzeuge nicht als einzelne Elemente simuliert werden, sondern immer als eine Menge von Fahrzeugen betrachtet werden (Weillkes, 2000, S.60), die nach Rose (2003, S.51) zu einem Element „verdampft“ werden. Die Modellansätze folgen Kontinuitätsgleichungen und werden durch Differentialgleichungen beschrieben (Rose, 2003, S.81). Nach Weillkes (2000, S.60) liegt die Schrittweite der Berechnungsschritte bei 10 Sekunden. Die erste makroskopische Verkehrssimulation ist auf Lighthill und Whitham (1955) zurückzuführen, welche Analogien zwischen der kinematischen Bewegung von Wellen in langen Flüssen und Verkehrsflüssen beschreiben.

Bei makroskopischen Verkehrssimulationen wird, wie von Kemper (2006, S.67ff) und Rose (2003, S.47f) angewendet, das topologische Streckennetzwerk als Graph beschrieben. Die Kanten modellieren Straßen mit einer definierten Kapazität an möglichen Verkehrsteilnehmern und einer grundlegenden Geometrie mit einer Länge, die für die Routenplanung verwendet wird. Die Knoten sind topologische Kreuzungspunkte, welche die Verknüpfungen der Kanten zueinander beschreiben. Nach Rose (2003, S.52) werden für die Kanten die mittlere Verkehrsdichte (Fahrzeuge pro Strecke), die durchschnittliche Geschwindigkeit oder der Verkehrsfluss (Fahrzeuge pro Zeit) auf den Streckenabschnitten definiert. Du u. a. (2004) beschreiben die makroskopische Ebene ebenfalls mit Knoten und Kanten beziehungsweise Verknüpfungen zwischen diesen Knoten. Des Weiteren werden die Kanten von Du u. a. (2004) mit dem Attribut der Anzahl der Fahrstreifen beschrieben. Dieses Attribut wird in der Verkehrssimulation erst auf einer mesoskopischen Ebene beschrieben.

Makroskopische Verkehrssimulationen werden vor allem bei der Planung und Auslegung von neuen Straßen eingesetzt. Weiterhin besteht die Möglichkeit, die Auswirkungen von Umplanungen von Routen oder Zuflussbeschränkungen von Straßen zu simulieren. Die makroskopischen Verkehrssimulationen haben den Vorteil, dass sie mit wenig detaillierten Informationen durchführbar sind. Allerdings lassen sich auf dieser Ebene beispielsweise keine detaillierten Kreuzungstopologien simulieren.

### 4.2.2 Mesoskopische Verkehrssimulation

Mesoskopische Verkehrssimulationen bilden die Verbindung zwischen der makroskopischen und mikroskopischen Verkehrssimulation. Die Modellierung findet ebenfalls auf Basis von Fahrzeugkollektiven statt, jedoch werden die einzelnen Fahrzeuge innerhalb der Kollektive simuliert (Weilkes, 2000, S.61f). Für die Durchführung der Verkehrssimulation werden Längsdynamikmodelle benötigt (Rose, 2003, S.100f). Das Straßennetzwerk wird nach Rose (2003, S.48f) in diskrete Segmente in Längsrichtung eingeteilt, wobei bei jeder Änderung der Fahrstreifenanzahl ein neues Segment beginnt. Damit wird die Anzahl der Fahrstreifen und deren Funktion in den jeweiligen Streckenabschnitten spezifiziert. Detailliertere Fahrstreifeninformationen, wie Markierungen oder Breiten, werden nicht modelliert. Die Verkehrsdichte wird aus den Fahrzeugen pro Segment berechnet. Du u. a. (2004) sehen auf der mesoskopischen Ebene den gleichen Detaillierungsgrad der Straße.

Rose (2003) zeigt in seiner Arbeit, wie aus makroskopischen und mikroskopischen Modellen mesoskopische Modelle berechnet werden. Mittels mesoskopischer Modellierung wird der Verkehrsfluss auf einer fahrstreifengenauen Abstraktion simuliert, wodurch die Möglichkeit besteht den Verkehrsfluss an Autobahnkreuzen oder Knotenpunkten beziehungsweise Engstellen fahrstreifengenau zu simulieren und zu untersuchen (Rose, 2003, S.102).

Die Vorteile gegenüber makroskopischen Verkehrssimulationen liegen in einer detaillierteren Darstellung der Straße durch die fahrstreifengenaue Modellierung und Simulation von einzelnen Fahrzeugen. Durch die Diskretisierung der Straße in Längs- und Querrichtung können jedoch keine kontinuierlichen Bewegungen innerhalb der Segmente dargestellt werden. Weiterhin ist es nicht möglich, das Querverhaltensverhalten der Fahrzeuge innerhalb eines Fahrstreifens zu simulieren.

### 4.2.3 Mikroskopische Verkehrssimulation

Mikroskopische Verkehrssimulationen modellieren das Verhalten von einzelnen Fahrzeugen. Dabei werden das Fahrzeug, der Fahrer sowie mögliche Assistenzsysteme oder automatisierte Fahrfunktionen als ein Element betrachtet (Weilkes, 2000, S.60). Die Parametrisierung der verwendeten Modelle beschreibt das individuelle Verhalten der einzelnen Verkehrsteilnehmer. Nach Weilkes (2000, S.60) sind die beschreibenden Größen Geschwindigkeits- und Wegverläufe beziehungsweise Zeitlücken zu anderen Verkehrsteilnehmern. Die Schrittweite der Simulation liegt bei einer Sekunde.

Für eine raumdiskrete Modellierung der Längsbewegung kann unter anderen das von Nagel und Schreckenberg (1992) vorgestellte Modell verwendet werden, welches zu der Gruppe der zellularen Automaten gehört. Für eine zeitkontinuierliche Modellierung der Längsbewegung können die von Treiber u. a. (2000) oder Wiedemann (1974) vorgestellten Fahrzeugfolgemodelle verwendet werden. Für die Querverführung und Fahrstreifenwechsel kann beispielsweise das Modell von Gipps (1986) verwendet werden. Mit diesem ist eine Simulation der Querverführung der individuellen Verkehrsteilnehmer innerhalb des Fahrstreifens möglich. Dadurch besteht die Möglichkeit, beispielsweise das Ausweichen um stationäre Objekte im Fahrstreifen zu simulieren.

Das Straßennetz wird durch Geometrien in einer quasi-raumkontinuierlichen Darstellung beschrieben. Fahrstreifen werden durch eine feste Breite und eine Markierung definiert. Du u. a. (2004) beschreiben die mikroskopische Ebene mit den gleichen Details. Beispiele für

mikroskopische Verkehrssimulationen sind unter anderen *SUMO*<sup>2</sup> (Krajzewicz u. a., 2012) oder *VISSIM*<sup>3</sup> (PTV Group).

Mikroskopische Verkehrssimulationen werden für die Untersuchung von längs- und querdynamischen Effekten von Verkehrsteilnehmern verwendet. Der Vorteil dieser Simulationsart liegt in der quasi-raumkontinuierlichen Beschreibung der Umgebung, wodurch die Möglichkeit besteht, das Verhalten der Verkehrsteilnehmer innerhalb eines Fahrstreifens oder bei Überholmanövern zu untersuchen.

#### 4.2.4 Submikroskopische Verkehrssimulation

Submikroskopische Verkehrssimulationen detaillieren die Teilkomponenten Fahrzeug und Umgebung. Nach Weilkes (2000, S.60) wird das Fahrzeug durch die Teilkomponenten Fahrer, das übrige Restfahrzeug und automatisierte Fahrfunktion modelliert. Das Restfahrzeug wird beispielsweise in die Komponenten Motor, Getriebe, Lenkung, etc. unterteilt. Die Straße wird unter anderem durch verschiedene Reibwerte und individuelle Fahrstreifenbreiten auf den verschiedenen Streckenabschnitten beschrieben.

Durch eine submikroskopische Verkehrssimulation besteht die Möglichkeit, die Interaktion zwischen den Teilkomponenten Fahrer, Fahrzeug und automatisierter Fahrfunktion zu analysieren. Eine submikroskopische Simulationssoftware ist unter anderem die von Von Neumann-Cosel (2013) und Dupuis u. a. (2010a) vorgestellte Software *Virtual Test Drive*, welche von der Firma *Vires*<sup>4</sup> weiterentwickelt wird. Diese simuliert das Verhalten der Fahrer und die Dynamik der Fahrzeuge durch unterschiedliche Simulationsmodelle. Zusätzlich können Assistenzsysteme oder automatisierte Fahrfunktionen integriert werden.

Nach Weilkes (2000, S.69) sind die Anwendungsgebiete der submikroskopischen Verkehrssimulation der Test von Assistenzfunktionen oder die Verbrauchsaussagen von Fahrzeugen. Die Schrittweite der Simulationen ist kleiner als 0,1 Sekunden. Die Vorteile liegen im hohen Detaillierungsgrad der Umgebung und des Fahrzeugs. Durch die geringe Schrittweite der Berechnungen gegenüber anderen Verkehrssimulationen kann ein erhöhter Rechenaufwand entstehen, wodurch erhöhte Anforderungen an die Simulationshardware gestellt werden oder eventuell die mögliche Simulationsgeschwindigkeit reduziert wird.

### 4.3 Klassifikation von X-in-the-Loop-Verfahren

X-in-the-Loop-Verfahren sind eine Möglichkeit, die Testdurchführung mit Hilfe von Simulationen durchzuführen. In diesem Kapitel werden verschiedene X-in-the-Loop-Verfahren anhand ihrer Komponenten klassifiziert. Dazu werden derzeit in der Literatur verwendete Klassifikationen beschrieben und eine neue Methode vorgestellt, die eine eindeutige Einteilung der X-in-the-Loop-Verfahren ermöglicht. Die Methode wird anschließend an verschiedenen X-in-the-Loop Verfahren angewendet. Die Klassifikation wird im modularen virtuellen Testbaukasten verwendet, um die erstellten Testfälle anhand von Gütekriterien den X-in-the-Loop-Verfahren zuzuordnen. Die Zuordnungsmethode wird in Kapitel 9.3 vorgestellt.

Teile dieses Kapitels wurden bereits in Schuldt u. a. (2015b)<sup>5</sup> veröffentlicht.

<sup>2</sup>SUMO steht für Simulation of Urban Mobility

<sup>3</sup>VISSIM steht für Verkehr in Städten-Simulation

<sup>4</sup><http://www.vires.com>, abgerufen: 30.06.2016

<sup>5</sup>Die Idee der Klassifikationsmethode durch Kiviat-Diagramme wurden vom Autor der vorliegenden Arbeit erbracht. Die Koautoren haben durch intensive Diskussionen zur Klassifikation der verschiedenen X-in-the-Loop-Verfahren beigetragen.

### 4.3.1 Stand der Technik

Die X-in-the-Loop-Verfahren werden in der Literatur anhand ihrer verschiedenen Komponenten klassifiziert. Eine Klassifikation von X-in-the-Loop-Verfahren wird von Strasser (2012, S.5ff, S.140) vorgestellt. Dieser klassifiziert die X-in-the-Loop-Verfahren anhand der Komponenten *Fahrzeugsystem*, *Fahrer*, *Fahrzeug*, *Umwelt* und *Verkehrsteilnehmer*. Zusätzlich wird das Testobjekt definiert und dessen Echtzeitfähigkeit bewertet. Die Verfahren *Software-in-the-Loop*, *Hardware-in-the-Loop*, *Driver-in-the-Loop* und *Vehicle-in-the-Loop* sowie der Realtest in Form von *Rapid Prototyping* und *On-Board-Test* werden klassifiziert. Der Unterschied zwischen den beiden Realtestverfahren liegt im Testobjekt. Während beim *Rapid Prototyping* der Code getestet wird, wird beim *On-Board-Test* das Steuergerät getestet. Für den Realisierungsgrad der Komponenten werden *simuliert* und *real vorhanden* vorgeschlagen. Diese Einteilung wird auch von Hartmann (2001, S.21ff) verwendet.

Von Neumann-Cosel (2013, S.43ff) beschreibt die X-in-the-Loop-Verfahren *Concept-in-the-Loop*, *Software-in-the-Loop*, *Hardware-in-the-Loop*, *Driver-in-the-Loop* und *Vehicle-in-the-Loop*. Die Verfahren werden klassifiziert anhand der Komponenten *Sensorik*, *Aktorik*, *Fahrdynamik*, *Umgebungsmodell* und *Fahrfunktion*. Dabei wird zwischen *simulierten* und *real vorhandenen* Komponenten unterschieden. Eine Stufe zur Realisierung der Komponenten durch ähnliche oder vergleichbare Ersatzkomponenten wird nicht eingeführt. Neben den X-in-the-Loop-Verfahren wird von von Neumann-Cosel (2013) ein *autonomes Erprobungsgelände* für automatisierte Fahrfunktionen erwähnt. Dabei handelt es sich um automatisiert fahrende Fahrzeuge, die mittels Fahrrobotern<sup>6</sup> definierte Trajektorien auf einem Testgelände abfahren.

Wachenfeld und Winner (2015a, S.460) führen als erstes eine Zwischenstufe, von ihnen *künstlich* genannt, zwischen den Stufen *simuliert* und *real vorhanden* für die Kategorien Umfeld und Fahrzeug ein. Der Realtest und das Verfahren *Software-in-the-Loop* werden klassifiziert. Weitere Verfahren werden nicht eingeteilt.

Durch die binäre Zuordnung der Dimensionen in die Stufen *simuliert* und *real vorhanden*, wie von Strasser, von Neumann-Cosel und Hartmann verwendet, lässt sich nicht jedes X-in-the-Loop-Verfahren eindeutig klassifizieren. Ein Beispiel sind Tests mit Ballonfahrzeugen<sup>7</sup>. Dabei werden die übrigen Verkehrsteilnehmer weder simuliert noch sind sie real vorhanden. Sie werden durch eine ähnliche Hardware mit ähnlichen Dimensionen repräsentiert. Diese Unterscheidung kann beim Test von Umfeldsensoren relevant sein, da die Sensordaten, die sich auf das Ballonfahrzeug beziehen, sich eventuell stark von Sensordaten echter oder simulierter Fahrzeuge unterscheiden. Daher wird im folgenden Abschnitt eine Klassifikationsmethode vorgestellt, die, wie von Wachenfeld und Winner (2015a, S.460) vorgeschlagen, eine Zwischenstufe zwischen *simuliert* und *real vorhanden* einführt. Mit dieser Methode werden anschließend verschiedene X-in-the-Loop-Verfahren klassifiziert. Weiterhin werden die Kategorien Fahrzeug und Umwelt weiter detailliert, um eine eindeutige Klassifikation der X-in-the-Loop-Verfahren zu ermöglichen.

### 4.3.2 Klassifikationsmethode für X-in-the-Loop-Verfahren

Die Klassifikation der X-in-the-Loop-Verfahren wird auf Basis von Kiviat-Diagrammen durchgeführt. Nach Morris und Roth (1982), zitiert nach Hoffmann (2008, S.270f), ermöglichen

<sup>6</sup>Ein nachträglich integriertes mechatronisches System, das die Pedalerie und das Lenkrad betätigt.

<sup>7</sup>Ballonfahrzeug: Ein aufblasbares Fahrzeug, welches die Dimensionen eines realen Fahrzeugs besitzt, jedoch durch Kollisionen nicht zerstört wird.



Kiviat-Diagramme eine übersichtliche Darstellung von mehrdimensionalen Beschreibungen. Mit den Diagrammen werden die verschiedenen Dimensionen der X-in-the-Loop-Verfahren visualisiert und zur Bewertung der einzelnen Eigenschaften verwendet. Die Diagramme werden durch die Anzahl der Achsen (Dimensionen) und die Parameterbereiche beziehungsweise Diskretisierungsstufen für jede Dimension klassifiziert. Die Dimensionen und Diskretisierungsstufen, die für die Beschreibung der unterschiedlichen X-in-the-Loop-Verfahren identifiziert wurden, werden im folgenden Abschnitt vorgestellt.

### Diskretisierungsstufen der Dimensionen

Die unterschiedlichen X-in-the-Loop-Verfahren substituieren verschiedene Systemkomponenten durch zeit- und kosteneffizientere sowie im Hinblick auf Sach- und Personenschäden risikoärmere Komponenten, die ein ähnliches Verhalten aufweisen. Die Ersatzkomponenten können das Verhalten der eigentlichen Systemkomponenten *simulieren* (lat. *simulatio* = Vortäuschung) oder *emulieren* (lat. *aemulatio* = Nacheiferung, Wetteifern). Die Begriffe der Simulation und Emulation wurden bereits in Kapitel 2.1.1 vorgestellt. Es wird die zusätzliche Stufe der *Emulation* eingeführt, da sich nach der Analyse der X-in-the-Loop-Verfahren einige Verfahren mit den Stufen *simuliert* und *real vorhanden* nicht eindeutig klassifizieren lassen.

Bei der Simulation wird das Verhalten der Systemkomponente vorgetäuscht. Die Ersatzkomponente hat hierbei keinerlei Hardwarebezug und wird ausschließlich in der Software beispielsweise durch Simulationsmodelle umgesetzt. Bei der Emulation wird das Verhalten der Systemkomponente durch ein, in Bezug auf die Zielhardware, vergleichbares Objekt nachgebildet, wie in Kapitel 2.1.1 beschrieben. Von außen wird das Objekt idealerweise funktionsgleich abgebildet. Beispiele sind Prozessor- oder Betriebssystememulatoren, die Programme oder Software für fremde Plattformen ausführen. Der Emulator gibt sich dabei als ein anderer Rechner aus<sup>8</sup>. Ein weiteres Beispiel ist ein Fahrroboter, welcher durch Bewegungen am Lenkrad einen realen Fahrer emuliert. Ein Fahrermodell, das den Lenkwinkel vorgibt, stellt hingegen eine Simulation des Fahrers dar.

Für die Klassifikation der X-in-the-Loop-Verfahren werden die Achsen der Kiviat-Diagramme mit einer nominalen Skala beschrieben, wobei die Skala in die Diskretisierungsstufen *simuliert*, *emuliert* und *real vorhanden* unterteilt wird. Einige Verfahren sind in der Lage, die Systemkomponenten in mehr als einer Dimension anzubieten. Dies wird in den Kiviat-Diagrammen durch Flächen veranschaulicht.

### Dimensionen zur Klassifikation von X-in-the-Loop-Verfahren

Für die Dimensionen der Kiviat-Diagramme wurden in Bagschik u. a. (2015a) als gemeinsame Systembestandteile der X-in-the-Loop-Verfahren die in Tabelle 4.1 dargestellten Dimensionen identifiziert. Ähnliche Dimensionen für die Einteilung von X-in-the-Loop-Verfahren in verschiedene Kategorien wurden von Strasser (2012, S.140) gewählt. Dieser nutzt als weitere Stufe das Fahrzeugsystem, welches hier allgemein Testobjekt genannt wird, um auch Teilelemente der automatischen Fahrfunktionen als Testgegenstand darstellen zu können. Die von Strasser (2012, S.140) genannte Umwelt wird unterteilt in die Szenerie und Verkehrsteilnehmer.

**Testobjekt:** Das Testobjekt beschreibt die Systemkomponenten, welche getestet werden, mitsamt der Ausführungsplattform. Im Falle einer automatisierten Fahrfunktion kann diese

<sup>8</sup><http://faql.de/wortgebrauch.html>, Aufgerufen: 20.02.2016

Dimension	Beschreibung
Testobjekt	Komponenten bzw. Hardware der automatisierten Fahrfunktion, die im Verfahren getestet wird
Fahrerverhalten	Beschreibung des Verhaltens des Fahrers am Lenkrad bzw. Mensch-Maschine-Schnittstelle
(Rest-)Fahrzeug	Elemente des Fahrzeugs, welche nicht zum Testobjekt gehören
Fahrdynamik	Bewegung des Fahrzeugs
(Rest-)Wahrnehmung	Elemente der Wahrnehmung, die nicht zum Testobjekt gehören
Verkehrsteilnehmer	Bewegliche Verkehrsteilnehmer um das Testobjekt
Szenerie	Stationäre Elemente um das Testobjekt

**Tabelle 4.1:** Übersicht der Dimensionen für die Einteilung der X-in-the-Loop-Verfahren nach (Bagschik u. a., 2015a)

real in Form eines Steuergerätes vorhanden sein, durch eine verwandte Hardware emuliert werden oder durch eine Entwicklungssoftware wie ADTF<sup>9</sup> oder Matlab<sup>10</sup> simuliert werden.

**Fahrerverhalten:** Das Fahrerverhalten beschreibt das Verhalten des Fahrers und seine Bedienung der Schnittstellen des Systems Fahrzeug, wie den Hebel eines Fahrtrichtungsanzeigers oder das Lenkrad. Das Fahrerverhalten kann real durch einen real existierenden Fahrer vorgegeben werden. Eine Emulation des Fahrerverhaltens stellt beispielsweise ein Fahrroboter dar, der das Lenkrad bewegt, um eine vordefinierte Trajektorie abzufahren oder aber einen Hebel mehrfach im Dauertest bedient. Eine Simulation des Fahrerverhaltens kann durch Fahrermodelle dargestellt werden.

**(Rest-)Fahrzeug:** Das (Rest-)Fahrzeug beschreibt die Elemente des Fahrzeugs, welche nicht Bestandteil des Testobjekts sind. Wenn ein reales Fahrzeug im Test verwendet wird, ist das (Rest-)Fahrzeug real vorhanden. Wenn das Zielfahrzeug nicht vorhanden ist und ein alternatives Fahrzeug oder eine ähnliche Hardware verwendet wird, wird das (Rest-)Fahrzeug emuliert. Bei einer Restbussimulation oder beim Software-in-the-Loop-Verfahren wird das (Rest-)Fahrzeug simuliert.

**Fahrdynamik:** Die Fahrdynamik beschreibt die Bewegung des Fahrzeugs. Sollte das Zielfahrzeug real vorhanden sein, ist die Fahrdynamik real vorhanden. Die Fahrdynamik wird emuliert, wenn ein anderes Fahrzeug, welches sich wie das Zielfahrzeug verhält, für den Test eingesetzt wird. Bei einer simulierten Fahrdynamik werden die Bewegungen des Fahrzeugs mittels eines Simulationsmodells berechnet und nicht in der Realität umgesetzt.

**(Rest-)Wahrnehmung:** Die (Rest-)Wahrnehmung beschreibt die Elemente der maschinellen Wahrnehmung, die nicht zum Testobjekt gehören. Die (Rest-)Wahrnehmung ist real, wenn die zugehörige Datenverarbeitung direkt auf der Zielplattform ausgeführt wird. Findet die Verarbeitung auf einer alternativen Hardwareplattform statt, beispielsweise einem Emulationsrechner mit einem Rapid-Prototyping-Framework, wird die (Rest-)Wahrnehmung

<sup>9</sup>ADTF: Automotive Data and Time-Triggered Framework, <http://www.audi-electronics-venture.de/aev/brand/de/leistungen/Entwicklungstools/adtf.html>, abgerufen: 20.02.2016

<sup>10</sup><http://de.mathworks.com/products/matlab/?requestedDomain=de.mathworks.com>, abgerufen: 20.02.2016

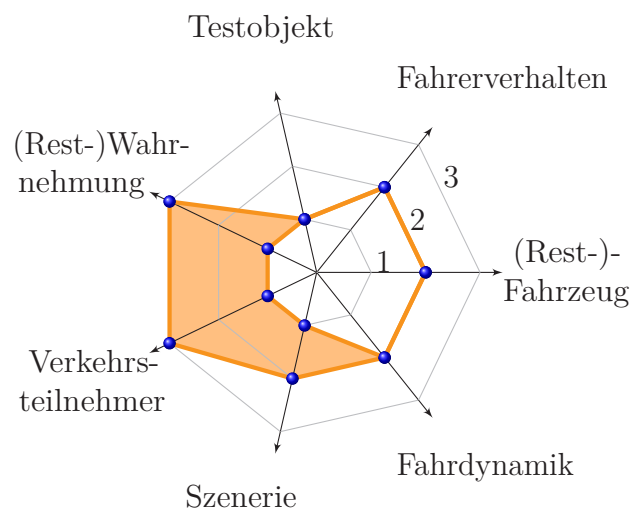


emuliert. Wenn künstlich erzeugte Objektlisten oder Sensordaten an die Funktion übermittelt werden, wird die (Rest-)Wahrnehmung simuliert.

**Verkehrsteilnehmer:** Die übrigen Verkehrsteilnehmer des Tests, die für einen Test benötigt werden, können ebenfalls real vorhanden sein, wenn beispielsweise die Tests auf einem Prüfgelände mit anderen Fahrzeugen durchgeführt werden. Beispiele für die Emulation der übrigen Verkehrsteilnehmer stellt der Einsatz von Ballon-Fahrzeugen, Crash-Targets oder einer Fußgängerbrücke dar. Eine Simulation der Verkehrsteilnehmer findet statt, wenn die Verkehrsteilnehmer ausschließlich in der Software existieren und dem Testobjekt vorgetäuscht werden.

**Szenerie:** Die Szenerie beschreibt das stationäre Umfeld um das Testobjekt. Die einzelnen Bestandteile der Szenerie wurden bereits in Kapitel 2.3 beschrieben. Finden die Tests im Feldversuch oder auf einem für den Test präparierten Testgelände statt, ist die Szenerie real vorhanden. Wird die Szenerie durch ähnliche Elemente dargestellt, findet eine Emulation der Szenerie statt. Beispiele sind künstliche Bordsteine oder Bäume. Sollten die Elemente durch eine Simulationssoftware generiert und dem Testobjekt vorgetäuscht werden, wird die Szenerie simuliert.

Abbildung 4.3 zeigt ein beispielhaft ausgefülltes Kiviat-Diagramm zur Klassifikation der X-in-the-Loop-Verfahren. Die Stufen der Dimensionen werden im Kiviat-Diagramm durch die Zahlen: 1 = simuliert, 2 = emuliert und 3 = real vorhanden gekennzeichnet. Im gewählten Beispiel wird das Testobjekt simuliert. Das Fahrerverhalten, das (Rest-)Fahrzeug und die Fahrdynamik werden emuliert. Die (Rest-)Wahrnehmung und die übrigen Verkehrsteilnehmer werden simuliert, emuliert oder sind real vorhanden. Dies wird durch die Fläche im Diagramm dargestellt. Die Szenerie wird simuliert oder emuliert.

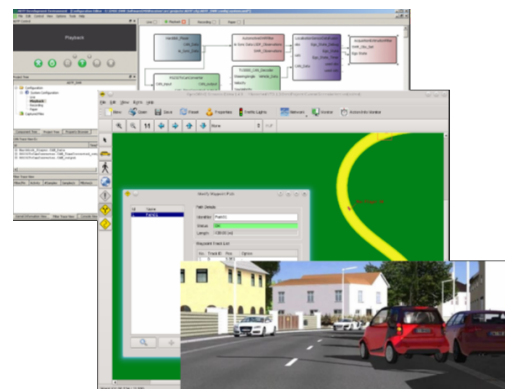
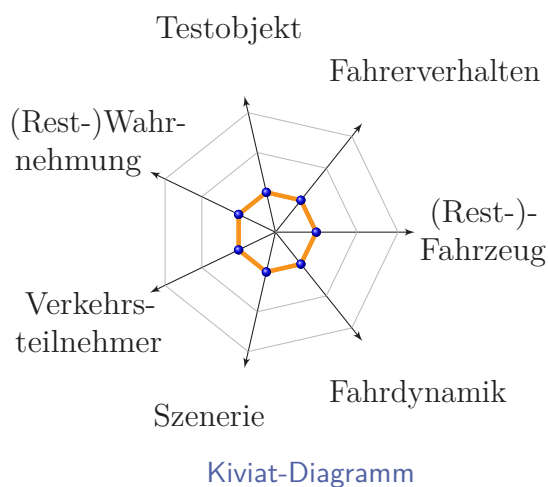


**Abbildung 4.3:** Beispielhaft ausgefülltes Kiviat-Diagramm zur Klassifikation von X-in-the-Loop Verfahren. Abkürzungen: 1 = simuliert, 2 = emuliert, 3 = real vorhanden.

### 4.3.3 Anwendung der Klassifikationsmethode auf verschiedene X-in-the-Loop-Verfahren

Im Folgenden wird die Klassifikationsmethode auf ausgewählte X-in-the-Loop-Verfahren angewendet. Neben den „klassischen“ X-in-the-Loop-Verfahren werden auch neuere Testwerkzeuge klassifiziert, die nicht mit X-in-the-Loop bezeichnet werden. Damit wird die Anwendbarkeit der Klassifikationsmethode auf diese neuen Testwerkzeuge gezeigt.

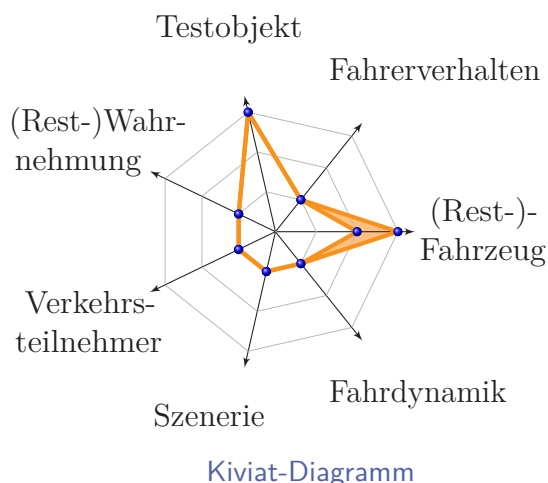
**Software-in-the-Loop:** Beim *Software-in-the-Loop* Verfahren werden sämtliche Elemente um das Testobjekt und das Testobjekt selbst simuliert, dargestellt in Abbildung 4.4.



Abbildung

**Abbildung 4.4:** Software-in-the-Loop. Links: Klassifikation im Kiviat-Diagramm. Rechts: Exemplarisches Bild einer Software-in-the-Loop Umgebung.

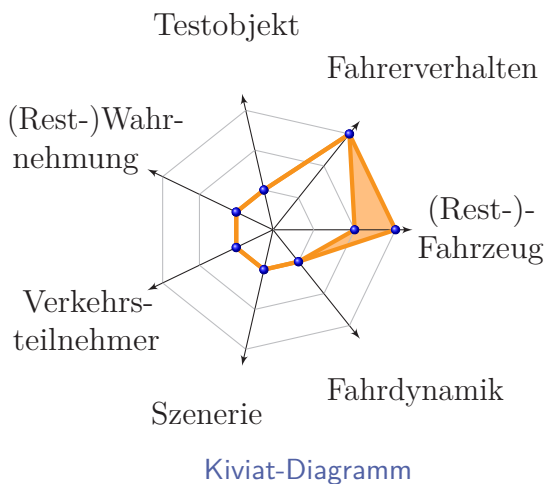
**Hardware-in-the-Loop:** Beim *Hardware-in-the-Loop* werden einzelne Steuergeräte getestet. Diesen werden mittels einer Restbussimulation die Signale der übrigen Steuergeräte vorgetäuscht. Alternativ werden verschiedene Steuergeräte im Verbund getestet. Dadurch ist das (Rest-)Fahrzeug real vorhanden oder wird emuliert, dargestellt in Abbildung 4.5. Die übrigen Dimensionen werden durch eine Restbussimulation simuliert.



Abbildung

**Abbildung 4.5:** Hardware-in-the-Loop. Links: Klassifikation im Kiviat-Diagramm. Rechts: Exemplarisches Bild eines Hardware-in-the-Loop Prüfstands.

**Driver-in-the-Loop:** Beim *Driver-in-the-Loop* Verfahren wird im Gegensatz zum *Software-in-the-Loop* das Fahrermodell durch einen realen Fahrer ersetzt, womit das Fahrerverhalten real ist. Tests, bei denen das *Driver-in-the-Loop* Verfahren angewendet wird, werden mit stationären oder beweglichen Simulatoren durchgeführt. Bei einem stationären Simulator wird die Fahrdynamik simuliert, während bei einem beweglichen Simulator die Fahrdynamik durch eine Bewegungsplattform emuliert wird. Das (Rest-)Fahrzeug wird bei einem stationären beziehungsweise beweglichen Simulator emuliert oder ist real vorhanden. Eine Emulation kann beispielsweise durch ein Lenkrad, wie in Abbildung 4.6 dargestellt, realisiert werden. Weiterhin besteht die Möglichkeit, reale Fahrzeuge in stationären oder beweglichen Simulatoren zu verwenden. Zeeb (2015) stellt einen solchen Simulator vor. Damit ist das (Rest-)Fahrzeug real vorhanden. Abbildung 4.6 zeigt das Ergebnis der Klassifikation von stationären Simulatoren. Abbildung 4.7 präsentiert das Ergebnis der Klassifikation für bewegliche Simulatoren.

Abbildung<sup>11</sup>

**Abbildung 4.6:** Driver-in-the-Loop mit einem stationären Simulator. Links: Klassifikation im Kiviat-Diagramm. Rechts: Exemplarisches Bild eines stationären Simulators.

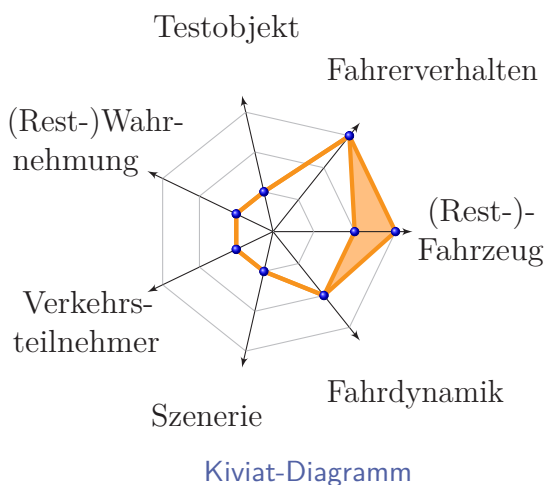


Abbildung (Quelle: © Daimler AG)

**Abbildung 4.7:** Driver-in-the-Loop mit einem beweglichen Simulator. Links: Klassifikation im Kiviat-Diagramm. Rechts: Exemplarisches Bild eines beweglichen Simulators.

<sup>11</sup>Bildquelle: [https://commons.wikimedia.org/wiki/File:Autohaus\\_Spaett\\_2008\\_5.JPG](https://commons.wikimedia.org/wiki/File:Autohaus_Spaett_2008_5.JPG), abgerufen 24.06.2016

**Vehicle-Hardware-in-the-Loop:** Beim *Vehicle-Hardware-in-the-Loop* wird ein reales Fahrzeug auf einem Rollenprüfstand fixiert (Verhoeff u. a., 2000; Hendriks u. a., 2010). Damit ist das (Rest-)Fahrzeug real vorhanden. Die Fahrdynamik wird durch den Rollenprüfstand emuliert, der Zustandsgrößen, wie die Steigung oder den Reibwert, der Straße verschieden nachbildet. Die Szenerie wird emuliert oder simuliert. Das Fahrerverhalten ist real vorhanden. Die übrigen Verkehrsteilnehmer werden simuliert, emuliert oder sind real vorhanden. Bei der Emulation ahmen Bewegungsplattformen die Verkehrsteilnehmer nach. Diese bewegen sich relativ zum Fahrzeug. Die übrigen Dimensionen werden simuliert, emuliert oder sind real vorhanden. Dies wird in Abbildung 4.8 durch Flächen dargestellt.

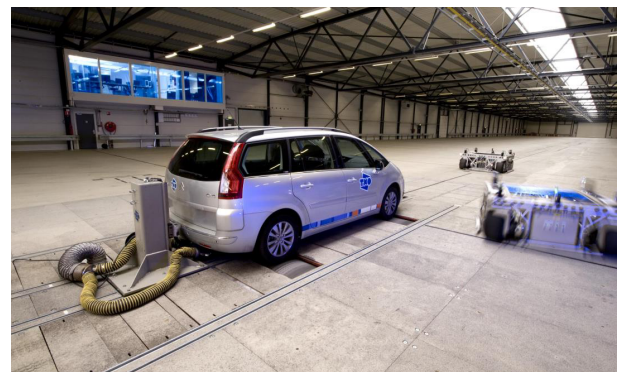
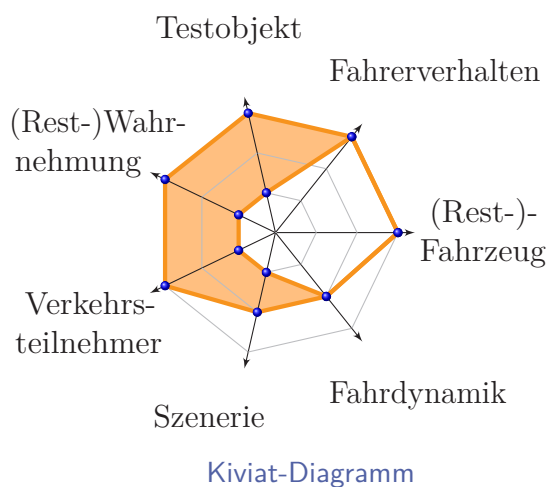


Abbildung (Quelle: © VeHIL Labor TNO, Helmond (NL))

**Abbildung 4.8:** Vehicle-Hardware-in-the-Loop. Links: Klassifikation im Kiviat-Diagramm. Rechts: Bild des Vehicle-Hardware-in-the-Loop Prüfstands.

**Vehicle-in-the-Loop:** Im Vergleich zum *Vehicle-Hardware-in-the-Loop* ist die Fahrdynamik beim *Vehicle-in-the-Loop* (Bock u. a., 2007; Bock, 2008) real vorhanden, da sich ein reales Fahrzeug auf einer Teststrecke bewegt. Damit ist auch das (Rest-)Fahrzeug real vorhanden. In der Version von Bock (2008) kann nur ein realer Fahrer das Fahrzeug fahren. Dadurch ist das Fahrerverhalten ebenfalls real vorhanden. Die übrigen Dimensionen können simuliert, emuliert oder real vorhanden sein, dargestellt in Abbildung 4.9 durch die Fläche.

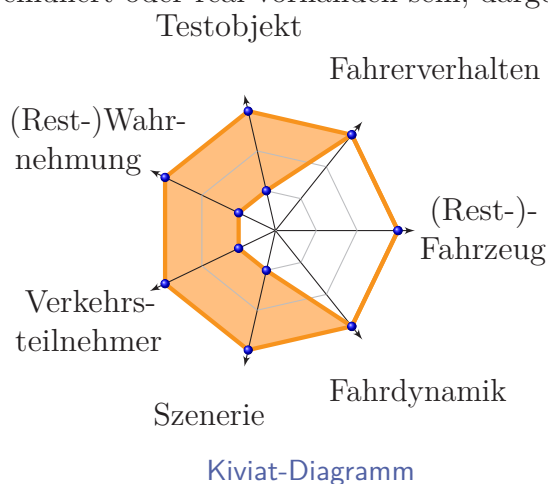


Abbildung (Quelle: Institut für Regelungstechnik, TU Braunschweig)

**Abbildung 4.9:** Vehicle-in-the-Loop. Links: Klassifikation im Kiviat-Diagramm. Rechts: Bild eines Vehicle-in-the-Loop Prüfstands.



**MOBILE:** Der Versuchsträger *MOBILE*, welcher von Bergmiller (2014, S.26ff) vorgestellt wird, stellt ein X-by-Wire Elektrofahrzeug dar, welches an allen vier Rädern über elektrischen Einzelradantrieb und elektrische Einzelradlenkung sowie elektromechanische Bremsen verfügt. Durch die redundanten Aktorikkomponenten werden verschiedene Ausprägungen von Fahrdynamikmodellen emuliert. Das Fahrerverhalten wird aufgrund der X-by-Wire Ansteuerung simuliert, emuliert oder ist real vorhanden. Durch prototypisch entwickelte Steuergeräte besteht die Möglichkeit, das Testobjekt in Hardware zu emulieren. Die (Rest-)Wahrnehmung wird simuliert, da aktuell keine Umfeldsensorik verbaut ist. Dadurch werden die Verkehrsteilnehmer ebenfalls simuliert. Die Szenerie ist hingegen real vorhanden. Abbildung 4.10 zeigt das ausgefüllte Kiviat-Diagramm.

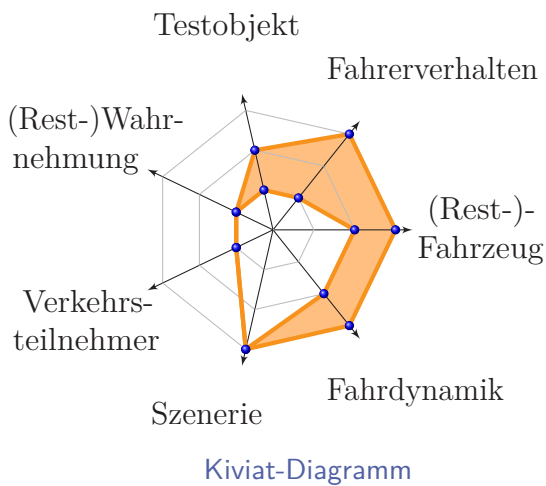


Abbildung (Quelle: Institut für Regelungstechnik, TU Braunschweig)

**Abbildung 4.10:** MOBILE. Links: Klassifikation im Kiviat-Diagramm. Rechts: Bild des Versuchsträgers MOBILE.

**InDrive:** Im Versuchsträger des Projekts *InDrive* (Gerson u. a., 2011) wird die Längsdynamik verschiedener Fahrzeuge in einem Passat CC emuliert. Damit wird die Fahrdynamik emuliert oder ist real vorhanden. Da das Testobjekt bei diesem Verfahren die Fahrdynamik darstellt, wird das Testobjekt ebenfalls emuliert oder ist real vorhanden. Die übrigen Elemente sind real vorhanden. Abbildung 4.11 zeigt das Ergebnis der Klassifikation.

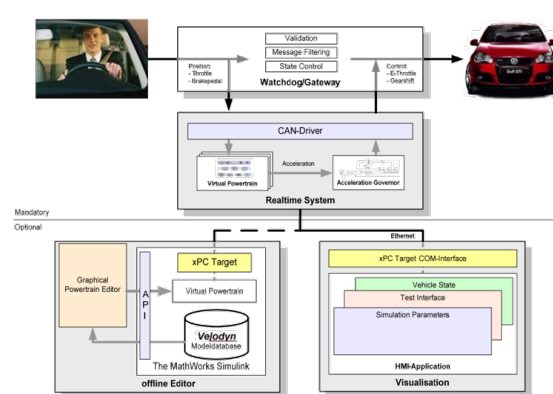
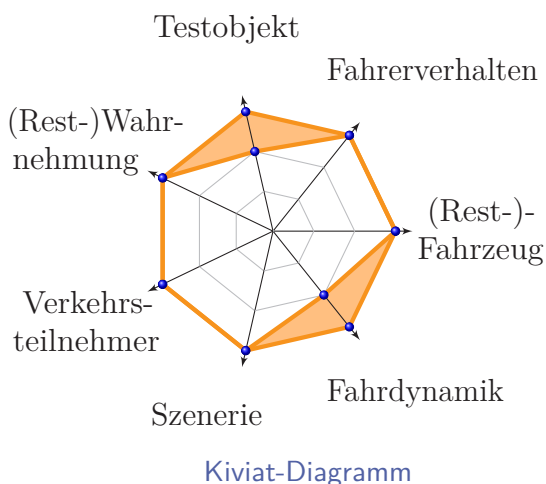


Abbildung (Quelle: Institut für Regelungstechnik, TU Braunschweig)

**Abbildung 4.11:** InDrive. Links: Klassifikation im Kiviat-Diagramm. Rechts: Schematische Darstellung des InDrive Projekts.

**Koordiniertes automatisiertes Fahren:** Beim *koordinierten automatisierten Fahren* der Firma Daimler (Schöner u. a., 2011) fahren automatisierte Versuchsträger über ein Testgelände und führen reproduzierbare Fahrmanöver aus. Das Lenkrad und die Pedalerie werden dabei durch einen Fahrroboter bewegt. Damit wird das Fahrerverhalten durch den Fahrroboter emuliert. Falls ein realer Fahrer das Lenkrad bewegt, ist das Fahrerverhalten real vorhanden. Das Testobjekt kann simuliert, emuliert oder real vorhanden sein. Die übrigen Dimensionen sind real vorhanden. Dies wird in Abbildung 4.12 dargestellt.

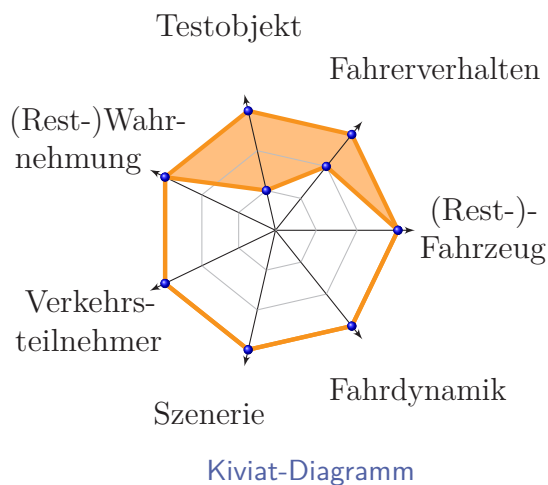
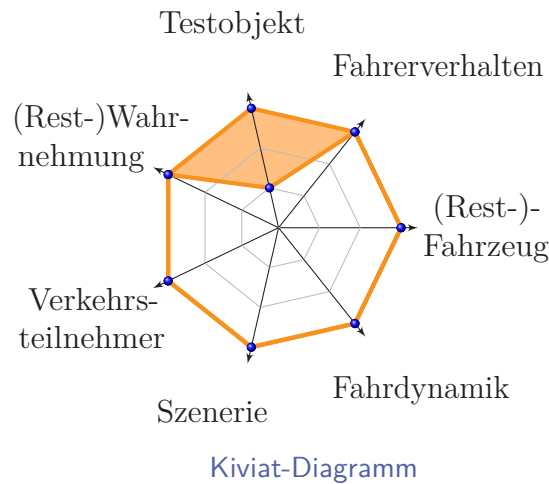


Abbildung (Quelle: © Daimler AG)

**Abbildung 4.12:** Koordiniertes automatisiertes Fahren. Links: Klassifikation im Kiviat-Diagramm. Rechts: Bild des Prüfgeländes inklusive Versuchsträger.

**Virtual Assessment of Automation in Field Operation:** Wachenfeld und Winner (2015b) beschreiben ein weiteres Testwerkzeug, um automatisierte Fahrfunktionen im Feldversuch zu testen. Dabei handelt es sich um eine Erweiterung des von Winner (2002) vorgestellten Ansatzes des *Trojanischen Pferdes*. Beim Ansatz von Winner (2002) werden der zu testenden Fahrfunktion im Fahrzeug alle benötigten Eingangssignale zur Verfügung gestellt. Die Ausgangssignale werden jedoch nicht an die Aktorik übergeben. Zur Bewertung des Systems werden die Ausgangssignale mit den Signalen eines Referenzsystems, wie zum Beispiel ein menschlicher Fahrer, verglichen. Dadurch ist ein Vergleich der aktuellen Ausgangssignale möglich. Jedoch ist es nicht möglich, die folgenden Auswirkungen der Fahrfunktion auf die Szene zu testen, da die Ausgangssignale nicht an die Aktorik übergeben werden und somit keine Rückkoppelung stattfindet. Bei der von Wachenfeld und Winner (2015b) beschriebenen Erweiterung werden die Eingangssignale verwendet, um eine Simulation mit der derzeitigen wahrgenommenen Szene zu initialisieren. In der Simulation werden die Ausgangssignale der zu testenden Fahrfunktion verarbeitet und daraus der weitere Verlauf der Szene in der Simulation berechnet. Zur Bewertung der Fahrfunktion wird das Verhalten der Fahrfunktion in der Simulation mit einem Referenzsystem verglichen. Wachenfeld und Winner (2015b) geben für das Beispiel einer Ausweich-Funktion als Referenzsystem einen menschlichen Fahrer mit optionalen Fahrerassistenzsystemen an. Da der Test im Feldversuch stattfindet, sind alle Dimensionen bis auf das Testobjekt real vorhanden. Je nach Realisierung der Fahrfunktion wird das Testobjekt simuliert, emuliert oder ist real vorhanden. Dies wird in Abbildung 4.13 dargestellt.



**Abbildung 4.13:** Virtual Assessment of Automation in Field Operation.

Durch die Klassifikation der „klassischen“ X-in-the-Loop-Verfahren und der neueren Testwerkzeuge *MOBILE*, *InDrive*, *koordiniertes automatisiertes Fahren* und *Virtual Assessment of Automation in Field Operation* wurde gezeigt, dass durch die Darstellung der Einteilung in Kiviat-Diagramme und durch die Einführung der Stufe der Emulation eine eindeutige Klassifikation der Verfahren möglich ist. Diese Klassifikation wird in Kapitel 9.3 für die Entwicklung einer Zuordnungsmethode für Testfälle auf X-in-the-Loop-Verfahren verwendet.

## 4.4 Zusammenfassung

In diesem Kapitel wurden Methoden des Softwaretests vorgestellt, die in Kapitel 8.4 für die systematische Ableitung von Testfällen für automatisierte Fahrfunktionen verwendet werden. Zudem wurden verschiedene Abstraktionsebenen der Verkehrssimulation präsentiert. Die verschiedenen Abstraktionsebenen werden in Kapitel 9.1.2 auf ihre Verwendbarkeit im modularen virtuellen Testbaukasten bewertet. Das Kapitel schließt mit einer neuen Methode für die Klassifikation von X-in-the-Loop-Verfahren ab. Durch die Klassifikation verschiedener „klassischer“ X-in-the-Loop-Verfahren und neuen Testwerkzeugen wurde gezeigt, dass durch die Einführung der Stufe der Emulation eine eindeutige Einteilung der Verfahren möglich ist. In Kapitel 9.3 wird diese Klassifikation für eine Zuordnungsmethode von Testfällen auf die X-in-the-Loop-Verfahren verwendet.



# **Teil III**

## **Der modulare virtuelle Testbaukasten**



## 5 Anforderungen an den modularen virtuellen Testbaukasten

Dieses Kapitel beschreibt die Anforderungen, die an den modularen virtuellen Testbaukasten gestellt werden, damit dieser ein effektives, effizientes und systematisches Testkonzept für automatisierte Fahrfunktionen darstellt. Zunächst werden allgemeine Anforderungen an das Testkonzept definiert. Diese orientieren sich an den von Wachenfeld und Winner (2015a, S. 447f) vorgestellten Anforderungen an ein Testkonzept für vollautomatisierte Fahrzeuge. Im Rahmen der vorliegenden Arbeit wurde die Formulierung der Anforderungen an ein Testkonzept parallel zu den Forschungsarbeiten zum systematischen Testen weiterentwickelt und detailliert. Weiterhin werden Anforderungen beschrieben, die von den Anwendern des Testbaukastens gestellt werden. Zusätzlich werden Anforderungen definiert, die durch die Testobjekte gestellt werden. Die Testobjekte werden hinsichtlich ihres Automatisierungsgrads und des Integrationsgrads in der funktionalen Systemarchitektur aus Kapitel 3.2 unterschieden.

### 5.1 Anforderungen an ein Testkonzept für automatisierte Fahrfunktionen

Die gestellten Anforderungen an das Testkonzept sind angelehnt an die von Wachenfeld und Winner (2015a, S. 447f) vorgestellten Anforderungen für ein effektives und effizientes Testkonzept für vollautomatisierte Fahrfunktionen. Die Begriffe der Effektivität und Effizienz wurden bereits in Kapitel 2.1 vorgestellt. Als Effektivitätskriterien werden von Wachenfeld und Winner (2015a, S. 447f) die Kriterien *repräsentativ - valide*, *variierbar* und *beobachtbar* genannt. Die Effizienzkriterien sind *ökonomisch*, *reproduzierbar*, *frühzeitig* und *sicher*. Im Folgenden werden die Anforderungen aufgegriffen und weiter detailliert. Weiterhin werden Anforderungen beschrieben, die erfüllt sein müssen, damit der Testbaukasten ein systematisches Testkonzept darstellt. In der vorliegenden Arbeit werden die Anforderungen durch die Systematik darin definiert, dass das Testkonzept in weitere systematische Entwicklungsprozesse integrierbar ist und den Anwendern ein geführtes Testkonzept in Form eines Prozessmodells angeboten wird. Dabei unterteilt das Prozessmodell den Test von automatisierten Fahrfunktionen in ein organisiertes Vorgehen, welches sich in verschiedene, strukturierte Abschnitte unterteilt, die wiederum entsprechende Methoden und Techniken für die Durchführung der Prozessschritte enthalten.

#### 5.1.1 Anforderungen durch Effektivitätskriterien

**Der Testbaukasten muss repräsentative und valide Testfälle für das Testobjekt ableiten.**

Für Wachenfeld und Winner (2015a, S.447) hat die Anforderung der Repräsentativität zwei Aspekte. Zum einen muss die Testfallerstellung dafür sorgen, dass die für die Freigabe benötigte Testabdeckung erreicht wird. Dies bedeutet, dass das Fahrzeug unter verschiedenen Umweltbedingungen getestet wird. Weiterhin muss die Testdurchführung den minimal

notwendigen Grad an Realität aufweisen. Dies impliziert nach Wachenfeld und Winner (2015a, S.447), dass die Vereinfachung der Realität weder das Verhalten des Testobjekts noch das Verhalten der Umwelt gegenüber dem realen Verhalten beeinflussen darf.

In der vorliegenden Arbeit wird die Repräsentativität ebenfalls auf die Testfallerstellung bezogen. Das Testkonzept muss Testfälle ableiten, die repräsentativ und valide für das Testobjekt sind. Dabei wird die Definition der Repräsentativität aus der Statistik verwendet. In der Statistik ist eine Stichprobe repräsentativ, wenn „sie die innere Struktur und die Charakteristika einer statistischen Grundgesamtheit adäquat abbildet“ (Eckstein, 2014, S.16). Übertragen auf die Testfallerstellung bedeutet dies, dass Parameter und Parameterwerte verwendet werden, welche in den Anwendungsgebieten des Testobjekts auftreten und zusätzlich die Anwendungsgebiete adäquat abbilden. Dies gilt sowohl für die beschreibenden Parameter der Umgebung, der beweglichen Elemente als auch der Umweltbedingungen. Damit besteht das Potential, repräsentative und valide Testfälle zu erstellen, um die von Wachenfeld und Winner (2015a, S.447) angesprochene benötigte Testabdeckung für die Freigabe des Systems zu erreichen. Um die Robustheit des Testobjekts zu überprüfen, sind zusätzlich invalide Testfälle durchzuführen. Dies bedeutet, dass Testfälle mit Parameterbereichen durchgeführt werden, die außerhalb der Spezifikation oder der Anwendungsbereiche liegen.

Des Weiteren wird von Wachenfeld und Winner (2015a, S.447), wie oben beschrieben, für die Repräsentativität des Testkonzepts ein minimal notwendiger Grad an Realität für die Testdurchführung gefordert, damit die Ergebnisse mit real durchgeführten Tests vergleichbar sind. Wie dieser minimal notwendige Grad an Realität definiert werden kann, stellt derzeit eine offene Forschungsfrage dar, die ebenfalls im Forschungsprojekt *PEGASUS*<sup>1</sup> (Plättner u. a., 2016) thematisiert wird. Für die Beantwortung der Forschungsfrage fehlen unter anderem Metriken für die Validierung und Gütebewertung der Simulationsmodelle. Eine Antwort auf diese Forschungsfrage wird in der vorliegenden Arbeit nicht gegeben, da keine Metriken für die Bewertung und Validierung der Simulationsmodelle entwickelt werden. An dieser Stelle sind weitere Forschungsarbeiten notwendig, um zu identifizieren, welche Güte für die Simulationsmodelle benötigt wird, damit ein Vergleich der Ergebnisse mit realen Tests möglich ist. In der vorliegenden Arbeit wird daher die Repräsentativität lediglich auf die Testfallableitung bezogen. In diesem Rahmen werden Herausforderungen diskutiert, die sich bei der Erstellung repräsentativer Testfälle ergeben.

### **Der Testbaukasten muss eine variierbare Testfallerstellung und Testdurchführung haben.**

Die Anforderung der Variation bedeutet nach Wachenfeld und Winner (2015a, S.448), dass die Testdurchführung alle von der Testfallerstellung abgeleiteten Testfälle mit den entsprechenden Parameterwerten durchführen kann.

In der vorliegenden Arbeit wird diese Anforderung in gleicher Form an die Testdurchführung des Testbaukastens gestellt. Weiterhin wird gefordert, dass die Methoden zur Auswahl und Analyse von repräsentativen Parametern und Wertebereiche so variabel sind, dass diese für verschiedene Verkehrsdomänen anwendbar sind. Des Weiteren muss in der Testfallerstellung die Szenario- beziehungsweise Testfallbeschreibung soweit variierbar sein, dass die identifizierten Parameter und Parameterwerte im Szenario und Testfall beschrieben werden können. Mit der Erfüllung dieser Anforderungen wird die Möglichkeit geschaffen,

---

<sup>1</sup>Projekt zur Etablierung von generell akzeptierten Gütekriterien, Werkzeugen und Methoden sowie Szenarien und Situationen zur Freigabe hochautomatisierter Fahrfunktionen

repräsentative Szenarien und Testfälle zu erstellen. Daher ergeben sich Abhängigkeiten zwischen der Anforderung der Repräsentativität und der Variabilität, da für die Erstellung von repräsentativen Testfällen die Variabilität der Szenario- und Testfallbeschreibung notwendig ist.

**Der Testbaukasten muss Messgrößen für die Testfallauswertung beobachtbar machen.**

Nach Wachenfeld und Winner (2015a, S.448) müssen für die Testfallauswertung die Parameter der Testdurchführung beobachtbar sein. Damit wird die Möglichkeit geschaffen, die Testfälle als *bestanden* oder *nicht bestanden* zu klassifizieren.

Diese Anforderung wird in der vorliegenden Arbeit in gleicher Weise an die Testdurchführung gestellt. Weiterhin wird für die Testfallauswertung gefordert, dass durch die beobachtbaren Parameter die Testfälle nicht nur als *bestanden* oder *nicht bestanden* klassifiziert werden können, sondern auch die Güte der Testergebnisse anhand von verschiedenen Metriken bewertbar ist. Mit der Erfüllung dieser Anforderung wird eine weitere Bewertung und Analyse der Testergebnisse ermöglicht.

### 5.1.2 Anforderungen durch Effizienzkriterien

**Der Testbaukasten muss eine ökonomische Testdurchführung ermöglichen.**

Nach Wachenfeld und Winner (2015a, S.448) ist diese Anforderung zweigeteilt. Zum einen soll die Testdurchführung möglichst schnell vorbereitet und durchführbar sein. Zum anderen wird gefordert, dass die Testdurchführung kostengünstig vorbereitet und durchgeführt wird.

Für den Testbaukasten werden diese Anforderungen ebenfalls auf die Ressourcen Zeit und Budget für die Testdurchführung bezogen. Um diese Anforderungen zu erfüllen, wird vom Testbaukasten gefordert, dass für die jeweiligen Tests das optimale Testwerkzeug hinsichtlich definierter Kriterien ausgewählt wird. Die Kriterien sind in erster Linie die oben genannten Ressourcen Zeit und Budget. Jedoch muss die Auswahlmethode soweit variabel sein, dass weitere Kriterien, wie die Modellgüte, hinzugefügt werden können. Damit wird die Möglichkeit geschaffen, eine ökonomische Testdurchführung basierend auf verschiedenen Kriterien zu realisieren.

**Der Testbaukasten muss reproduzierbar Testfälle durchführen können.**

Das Testkonzept muss nach Wachenfeld und Winner (2015a, S.448) reproduzierbare Testergebnisse liefern, um so den Aufwand für Regressionstests zu minimieren.

Vom Testbaukasten wird ebenfalls eine Reproduzierbarkeit der Testergebnisse gefordert. Weiterhin wird eine Reproduzierbarkeit bezogen auf den Ablauf des Szenarios und das Verhalten der übrigen Verkehrsteilnehmer gefordert. Dies gilt insbesondere bei Szenarien mit beweglichen beziehungsweise sich nicht deterministisch verhaltenden Elementen, wie einem realen Fahrer. Es wird zwischen der Reproduzierbarkeit aus globaler Sicht und der Reproduzierbarkeit aus Sicht des Testobjekts unterschieden. Bei der globalen Reproduzierbarkeit verhalten sich die übrigen Verkehrsteilnehmer unabhängig vom Verhalten des Testobjekts. Damit ist das Verhalten der übrigen Verkehrsteilnehmer, beispielsweise die Position zu definierten Zeitpunkten, aus globaler Sicht bei jeder Testdurchführung identisch. Bei einer Reproduzierbarkeit aus Sicht des Testobjekts adaptieren die übrigen Verkehrsteilnehmer ihr Verhalten, damit eine gewünschte Zielszene aus Sicht des Testobjekts reproduzierbar erreicht wird. Wenn sich das Testobjekt reproduzierbar verhält, treten beide Formen der Reprodu-

zierbarkeit gleichzeitig auf. Durch die Erfüllung dieser Anforderung wird die Möglichkeit geschaffen, die gewünschte Zielszene mit beweglichen Elementen in jeder Testdurchführung zu erreichen.

**Der Testbaukasten muss frühzeitig im Entwicklungsprozess einsetzbar sein.**

Das Testkonzept muss nach Wachenfeld und Winner (2015a, S.448) frühzeitig im Entwicklungsprozess einsetzbar sein, damit Fehler frühzeitig aufgedeckt werden können und zur Behebung der Fehler weniger Entwicklungsschritte durchlaufen werden müssen.

Diese Anforderung wird in gleicher Form an den Testbaukasten gestellt. Durch den frühen Einsatz des Testkonzepts können Fehler frühzeitig entdeckt und ökonomisch behoben werden. Damit muss der Testbaukasten vor allem Komponententests und Integrationstests aber auch Systemtests, wie in Kapitel 4.1.2 vorgestellt, ermöglichen, um frühzeitig im Entwicklungsprozess einsetzbar zu sein.

**Der Testbaukasten muss eine sichere Testdurchführung zur Verfügung stellen.**

Nach Wachenfeld und Winner (2015a, S.448) soll das Risiko der Gefährdung der Testobjekte und aller Testbeteiligten nicht das akzeptierte Risiko der Testdurchführung übersteigen. Dies gilt insbesondere bei der Durchführung von Realtests.

Diese Anforderung wird ebenfalls an den Testbaukasten gestellt. Das Testkonzept soll ein zerstörungsfreies Testen und gleichzeitig eine hohe Ergebnisgüte ermöglichen. Daher wird an dieser Stelle gefordert, dass der Testbaukasten bei der Auswahl eines Testwerkzeugs unterstützt und diese Anforderung dabei beachtet. Das Testwerkzeug sollte unter der Bedingung der Sicherheit des Testobjekts und der Testbeteiligten eine optimale Ergebnisgüte erreichen. Diese Anforderung ist besonders bei sicherheitsrelevanten Fahrfunktionen von Bedeutung, da diese Funktionen nach Funktionsdefinition erst auslösen oder eingreifen, wenn die Sicherheit der Insassen gefährdet ist. Ein Beispiel für ein solches System ist die von Reschka u. a. (2015b) beschriebene automatische Notbremsung, die erst im Notfall eingreift, um den Unfall zu verhindern oder die Unfallschwere zu mindern.

### 5.1.3 Anforderungen durch Systematik

**Der Testbaukasten soll in vorhandene systematische Entwicklungsprozesse integrierbar sein.**

Der Testbaukasten soll in vorhandene systematische Entwicklungsprozesse für automatisierte Fahrfunktionen integrierbar sein. Ein solcher systematischer Entwicklungsprozess wird beispielsweise von Reschka u. a. (2015b, S.927) skizziert. Der Prozess wird durch eine kleine und größere Iterationsschleife beschrieben. In der kleinen Iteration werden die Prozessschritte *Motivation durch Kundennutzen*, *Funktionsdefinition*, *technische Machbarkeit*, *Produktsicherheit*, *Mensch* und *Markt* durchgeführt und bewertet. Sollte die Iteration zu einem Ergebnis kommen, welches eine weitere Entwicklung befürwortet, werden die Schritte *Package*, *Systemarchitektur*, *Validierung in Test und Simulation*, *Sicherheitskonzept* und abschließende *Funktionsdefinition/-spezifikation* durchgeführt. Sollte die Iteration zu einem Ergebnis kommen, welches für eine Änderung der Funktion spricht, wird die kleine Iteration wiederholt. Der Testbaukasten soll sich in den von Reschka u. a. (2015b, S.927) beschriebenen Entwicklungsprozess integrieren lassen. Damit wird die Möglichkeit geschaffen, den Testbaukasten in einer systematischen Gesamtentwicklung von automatisierten Fahrfunktionen zu verwenden.

**Der Testbaukasten muss eine systematische Erstellung, Durchführung und Auswertung von Testfällen ermöglichen.**

Vom Testkonzept wird gefordert, dass die Tests systematisch erstellt, durchgeführt und ausgewertet werden. Hierfür muss das Testkonzept durch ein strukturiertes Prozessmodell mit einer festen Sequenz von Schritten beschreibbar sein. Der Benutzer ist bei der Anwendung gezielt durch die Prozessschritte der Erstellung, Durchführung und Auswertung der Tests zu führen. Weiterhin müssen verschiedene Methoden im Testkonzept integrierbar sein, damit die Möglichkeit besteht, das Prozessmodell an verschiedene Testobjekte anzupassen. Dies bedeutet zugleich, dass das Testkonzept die Eingaben und Ausgaben seiner Module auf einheitliche Schnittstellen normiert, damit die Methoden innerhalb der Module austauschbar sind.

## 5.2 Anforderungen durch Anwender

Neben den zuvor beschriebenen allgemeinen Anforderungen an ein effizientes, effektives und systematisches Testkonzept werden weitere Anforderungen durch die Anwender des modularen virtuellen Testbaukastens gestellt. Diese werden im Folgenden vorgestellt.

**Der Testbaukasten muss standardisierte Beschreibungsformate verwenden.**

Vom Testkonzept wird gefordert, dass standardisierte Formate für die Beschreibung der Einflussparameter, der Szenarien, der Testfälle sowie für die Auswertung verwendet werden. Dies ist unter anderem relevant für die Durchführung der Testfälle mit verschiedenen Testwerkzeugen. Die Testfälle sollen so beschrieben sein, dass sie gleichermaßen in der Simulation als auch im Realtest durchführbar sind. Dies gilt ebenfalls für die Beschreibung der Szenarien, der Szenarioparameter sowie für die Bewertungskriterien.

**Der Testbaukasten muss eine manuell und automatisch anpassbare Testfallerstellung ermöglichen.**

Die Anwender müssen einen Zugriff auf die Testfallerstellung und auf die einzelnen Parameter der Testfälle haben, damit eine manuelle Anpassung der Testfälle und eine Variation der Einflussparameter möglich ist. Dies ist beispielsweise notwendig, um einen Testfall soweit zu verändern, bis dieser gerade noch bestanden oder gerade nicht mehr bestanden wird, um so die Grenzen des Systems zu identifizieren. Damit wird eine manuelle Erstellung neuer Testfälle oder die Anpassung existierender Testfälle ermöglicht. Weiterhin soll dieser Schritt automatisch durchführbar sein. Damit wird die Möglichkeit geschaffen, aus existierenden Testergebnissen neue Testfälle zu erstellen, um automatisch die Testfälle zu identifizieren, bei denen das System am „schlechtesten“ oder „besten“ funktioniert.

**Der Testbaukasten muss die Testergebnisse automatisch dokumentieren.**

Die Testergebnisse sollen automatisch dokumentiert werden. Zusätzlich soll das getestete Szenario und die verwendete Testdurchführung zu den Testergebnissen abgelegt werden. Die Messdaten und die Szenarien sollen weiterhin mit definierten Markern gekennzeichnet werden, um die Ergebnisse für weitere Auswertungen gezielt suchen zu können.

## 5.3 Anforderungen durch Testobjekte

Die Anforderungen, die durch das Testobjekt gestellt werden, werden nach dem Integrationsgrad des Testobjekts in der funktionalen Systemarchitektur nach Kapitel 3.2 und



dem Automatisierungsgrad nach Kapitel 3.1 unterteilt. Dabei beschreibt der Begriff Integrationsgrad in der vorliegenden Arbeit, inwiefern das Testobjekt in der funktionalen Systemarchitektur integriert ist, und damit, welche Module der Systemarchitektur vom Testobjekt dargestellt werden.

### **5.3.1 Integrationsgrad des Testobjekts in der funktionalen Systemarchitektur**

Je nach Integrationsgrad des Testobjekts in der funktionalen Systemarchitektur ergeben sich unterschiedliche Anforderungen an das Testkonzept. Dies gilt besonders für die Testdurchführung, da dort verschiedene Testtreiber, wie sie in Kapitel 9.4 vorgestellt werden, für die Durchführung der Tests benötigt werden. Eine weitere Betrachtung der Anforderungen wird an dieser Stelle nicht durchgeführt, da die Anforderungen und die Herausforderungen, die sich durch den Integrationsgrad des Testobjekts ergeben, in Kapitel 9.1.1 im Detail diskutiert werden. An dieser Stelle sei nur darauf hingewiesen, dass die Anforderungen durch den Integrationsgrad des Testobjekts bereits an dieser Stelle bedacht wurden.

### **5.3.2 Anforderungen durch verschiedene Automatisierungsgrade**

Durch die verschiedenen Automatisierungsgrade der Fahrfunktionen ergeben sich weitere Anforderungen an das Testkonzept. Zur Diskussion der Anforderungen werden die Automatisierungsgrade 0 - 2 und die Automatisierungsgrade 3 - 5 gemeinsam betrachtet.

#### **Automatisierungsgrade 0 - 2: Keine Automation bis Teilautomatisierung**

Bei den Automatisierungsgraden 0 - 2 stellt der Fahrer die Rückfallebene für die automatisierte Fahrfunktion dar. Der Fahrer muss das System dauerhaft überwachen und jederzeit die Fahrzeugführung übernehmen können. Daher sollten in erster Linie Tests an den Systemgrenzen durchgeführt werden, um die Beherrschbarkeit der Übernahme der Fahrzeugführung zu testen. Daraus ergibt sich die Anforderung, dass das Testkonzept verschiedene Fahrermodelle zur Verfügung stellen muss, um diese Szenarien zu testen.

Die Systeme können jeder Zeit deaktiviert und innerhalb der Systemgrenzen aktiviert werden. Daher ist es möglich, das System für den Test in sämtliche Zielszenen zu bringen, solange sich diese Zielszenen innerhalb der Systemgrenzen befinden. Um die Möglichkeit zu schaffen, das System in die Zielszene zu bringen, wird eine hohe Variabilität und Reproduzierbarkeit vom Testkonzept gefordert. Dadurch werden die schon beschriebenen Anforderungen durch die Automatisierungsgrade 0 - 2 erneut gestellt.

#### **Automatisierungsgrade 3 - 5: Bedingte Automatisierung bis Vollautomatisierung**

Beim Automatisierungsgrad 3 stellt der Fahrer die Rückfallebene für das System dar. Der Fahrer muss das System allerdings nicht dauerhaft überwachen und hat für die Übernahme der Fahrzeugführung eine definierte Zeitspanne zur Verfügung. Da diese Zeitspanne nach Damböck (2013, S.148) bis zu 8 Sekunden beträgt, wird in der vorliegenden Arbeit davon ausgegangen, dass das System ein Großteil der in dieser Zeit auftretenden Szenarien von selbst lösen muss. Bei den Automatisierungsgraden 4 und 5 besitzt das System keine andere Rückfallebene, wodurch automatisierte Fahrfunktionen mit diesen Automatisierungsgraden in allen Anwendungsszenarien getestet werden müssen.

Da Systeme mit den Automatisierungsgraden 3 - 5 die Überwachung der Fahrumgebung übernehmen, sind sie nicht in jeder Szene aktivierbar. Die Systeme mit Automatisierungsgrad 4 sind dauerhaft in allen Fahrsituationen aktiv, für die die Systeme ausgelegt sind. Da die Systeme mit Automatisierungsgrad 5 für alle Fahrsituationen ausgelegt sind, sind diese dauerhaft aktiv und können nicht deaktiviert werden, um beispielsweise den Start eines Testfalls in einer definierten Szene zu ermöglichen. Damit können die Systeme mit den Automatisierungsgraden 3 - 5 womöglich im Test nicht in die gewünschte Zielszene gebracht werden, da die Systeme entweder in den Zielszenen nicht aktivierbar sind oder das Erreichen der Zielszene aktiv vom System verhindert wird.

Mit einer gezielten Variation des Umfelds, während der Testdurchführung und währenddessen das System aktiv ist, könnte die Zielszene womöglich dennoch erreicht werden. Dabei kann jedoch der Fall eintreten, dass mit realistischen Variationen der Umgebung ein Erreichen der Zielszene ebenfalls nicht möglich ist, da die automatisierte Fahrfunktion aktiv verhindert, die Zielszene zu erreichen (Maurer und Reschka, 2015). Ein Beispiel ist das zu dichte Auffahren auf ein bewegliches Element, da die Umfeldwahrnehmung das Element zu spät wahrnimmt oder die Detektion von nicht existierenden Elementen durch die Umfeldwahrnehmung. Daher ergeben sich an dieser Stelle besondere Anforderungen an die Variabilität der Testdurchführung, die eventuell nur von der Simulation erfüllt werden können, da dort Variationen des Umfelds möglich sind, die in der Realität unmöglich sind. Ein Beispiel ist das plötzliche und kurzzeitige Erscheinen von neuen beweglichen Elementen, um die Detektion von nicht existierenden Elementen zu testen.

Damit kann die Vermutung aufgestellt werden, dass für eine Absicherung von Fahrfunktionen mit den Automatisierungsgraden 3 - 5 die Simulation benötigt wird. In der Simulation besteht die Möglichkeit, die Umgebung soweit zu verändern, dass die für die Tests benötigten Zielszenen mit Hilfe von zum Teil unrealistischen Veränderungen der Umgebung erreicht werden. Für das Testkonzept bedeutet dies die Bestätigung der Anforderungen der Variabilität und Reproduzierbarkeit für die Simulation.



## 6 Aufbau und Schnittstellen des modularen virtuellen Testbaukastens

Dieses Kapitel beschreibt den Aufbau des modularen virtuellen Testbaukastens als Prozessmodell. Dabei unterteilt das Prozessmodell den Test von automatisierten Fahrfunktionen in ein organisiertes Vorgehen, welches sich in verschiedene, strukturierte Module unterteilt, die wiederum entsprechende Methoden und Techniken für die Durchführung der Prozessschritte enthalten. Darauf aufbauend werden die Schnittstellen zwischen den Modulen des Testbaukastens beschrieben.

### 6.1 Aufbau des Testbaukastens

Der modulare virtuelle Testbaukasten ist ein mehrschichtiges Konzept für den systematischen, funktionalen Test von automatisierten Fahrfunktionen in virtuellen Umgebungen. Dieser verfolgt das Ziel, effizient und effektiv Testfälle für automatisierte Fahrfunktionen zu erstellen, durchzuführen und auszuwerten. Abbildung 6.1 zeigt den Aufbau des Testbaukastens. Mit Hilfe des Testbaukastens werden die automatisierten Fahrfunktionen mittels der Durchführung der Module *Auswahl und Analyse von Einflussparametern*, *Testfallerstellung*, *Testdurchführung* und *Testfallauswertung* getestet. Die vier Module des Testbaukastens werden in den folgenden Kapiteln im Detail beschrieben.

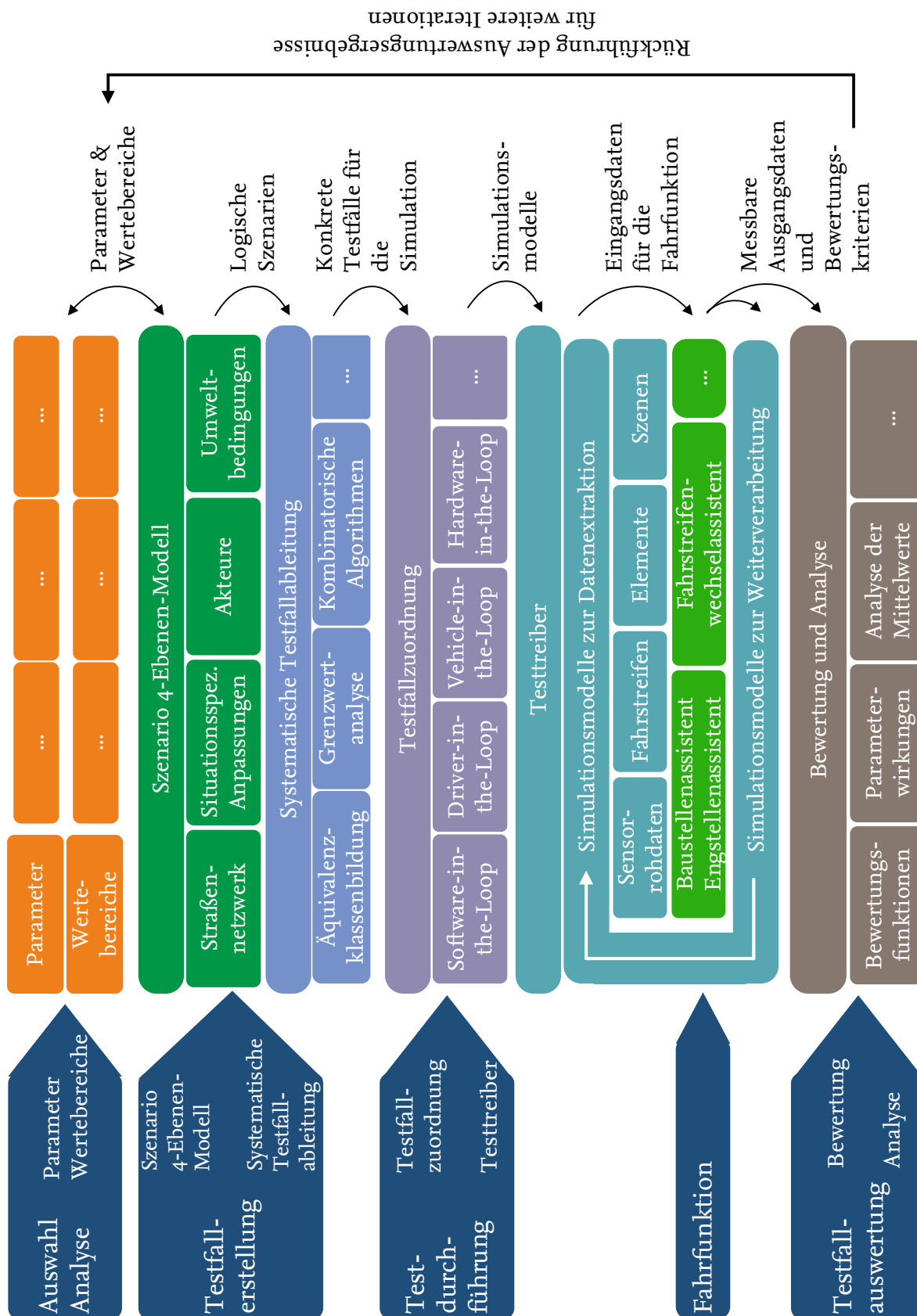
Ein weiteres Ziel des Testbaukastens ist die Sammlung von Testmethoden für die zuvor genannten Module, wodurch ein Methodenbaukasten entsteht. Der Fokus liegt auf funktionalen Prüfmethoden, wie sie in Kapitel 4.1 vorgestellt wurden. Durch die Definition von einheitlichen Schnittstellen zwischen den Modulen werden die Methoden der jeweiligen Module austauschbar. Damit wird die Möglichkeit geschaffen, eine Anpassung des Testkonzepts auf die jeweiligen Testobjekte durch die Auswahl von Methoden zu ermöglichen.

Im ersten Modul findet eine Auswahl und Analyse von Einflussparametern statt. Diese können aus verschiedensten Informationsquellen abgeleitet werden. Zusätzlich werden Wertebereiche und Diskretisierungsstufen für die einzelnen Einflussparameter identifiziert.

Im zweiten Modul werden die Szenarien und Testfälle für den Test der automatisierten Fahrfunktion mit Hilfe eines 4-Ebenen-Modells, welches in Kapitel 8.3 vorgestellt wird, und einer systematischen Testfallableitung, welche in Kapitel 8.4 präsentiert wird, erstellt. Für die Anwendung des Testbaukastens wird die Annahme getroffen, dass der Quellcode des Testobjekts nicht für die Testfallableitung zur Verfügung steht. Damit werden für die systematische Testfallableitung lediglich Methoden des Black-Box-Testens für die Erstellung von konkreten Testfällen aus den Szenarien verwendet.

Darauf aufbauend werden die erstellten Testfälle mittels einer Zuordnungsmethode auf die in Kapitel 4.3 klassifizierten Methoden zur Testdurchführung verteilt und anschließend mit diesen durchgeführt. Mit Hilfe von verschiedenen Testtreibern werden zum einen die Eingangsdaten für das Testobjekt zur Verfügung gestellt und zum anderen die Ausgangsdaten des Testobjekts für die Berechnung neuer Eingangsdaten verwendet.

Im vierten Modul werden die Testergebnisse anhand von verschiedenen Kriterien bewertet und analysiert. Ziel ist die Identifikation der Wirkungen von verschiedenen Eingangsparametern.



**Abbildung 6.1:** Aufbau, Schnittstellen und Bestandteile der verschiedenen Module des modularen virtuellen Testbaukastens. Die Module werden im Detail in den folgenden Kapiteln diskutiert.

tern auf die Bewertungskriterien. Dazu werden statistische Bewertungsverfahren verwendet. Die Ergebnisse der Testfallauswertung können wiederum im Modul der Auswahl und Analyse genutzt werden, um weitere Einflussparameter und Diskretisierungsstufen zu identifizieren. Dadurch wird die Möglichkeit geschaffen, den Testbaukasten iterativ zu verwenden.

## 6.2 Schnittstellen zwischen den Modulen

Dieses Unterkapitel beschreibt die Schnittstellen zwischen den Modulen des modularen virtuellen Testbaukastens. Dazu werden die Funktion des Moduls sowie die Eingangs- als auch Ausgangsschnittstellen beschrieben. Eine detaillierte Beschreibung, Analyse und Diskussion der einzelnen Module findet in den folgenden Kapiteln statt.

### Modul: Auswahl und Analyse

**Kurzbeschreibung:** Im Modul der Auswahl und Analyse wird eine Auswahl von Einflussparametern getroffen und diese Auswahl anschließend analysiert, um Wertebereiche und Diskretisierungsstufen für die Testfallerstellung zu definieren. Das Modul der Auswahl und Analyse inklusive der sich ergebenden Herausforderungen bei der Auswahl und Analyse von Einflussparametern werden in Kapitel 7 detailliert vorgestellt.

**Eingangsschnittstellen:** Eingangsschnittstellen für die Auswahl sind Funktionsdefinitionen, Systemanforderungen, existierende Szenarienkataloge, geltende Richtlinien und Normen und weitere Informationsquellen, wie eventuell vorhandene Testergebnisse aus vorherigen Testfallauswertungen. Die verschiedenen Informationsquellen werden in Kapitel 7.3 bewertet. Die Analyse nutzt als Informationsquellen existierende Statistiken, Expertenwissen, existierende Normen und Richtlinien sowie vorgeschriebene Grenzwerte für identifizierte Parameter. Weiterhin werden vorhandene Testergebnisse zur Optimierung der Analyse verwendet. Dadurch bietet der Testbaukasten die Möglichkeit, die automatisierten Fahrfunktionen in mehreren Iterationen zu testen. Die Informationsquellen zur Analyse der Testergebnisse werden in Kapitel 7.4 bewertet. Zusätzlich können Parameter, die bereits im 4-Ebenen-Modell zur Beschreibung von Szenarien identifiziert wurden, für die Identifikation von Einflussparametern für die automatisierte Fahrfunktion hinzugezogen werden.

**Ausgangsschnittstellen:** Die Ausgangsschnittstellen sind identifizierte Einflussparameter für die Fahrfunktion und die Umgebung, in der die Fahrfunktion agiert. Für jeden Parameter werden sowohl gültige als auch ungültige Wertebereiche sowie Diskretisierungsstufen für die jeweiligen Wertebereiche definiert.

### Modul: Testfallerstellung

**Kurzbeschreibung:** Die Testfallerstellung beinhaltet die Module der Szenariobeschreibung anhand des 4-Ebenen-Modells, welches in Kapitel 8.3 vorgestellt wird, sowie der systematischen Testfallableitung, welche in Kapitel 8.4 detailliert beschrieben wird. Mittels der systematischen Testfallableitung werden aus den Szenarien Testfälle abgeleitet.

**Eingangsschnittstelle:** Über die Eingangsschnittstelle werden die identifizierten Parameter des Moduls der Auswahl und Analyse an das 4-Ebenen-Modell übergeben. Mittels der identifizierten Parameter werden die Ebenen des Modells befüllt. Damit werden logische Szenarien erstellt, wie sie in Kapitel 2.3 vorgestellt wurden. Logische Szenarien stellen die Schnittstelle zur Testfallableitung dar. Aus jedem logischen Szenario werden mittels der systematischen Testfallableitung konkrete Testfälle mit festgelegten Parameterwerten abgeleitet.

Um die Testfälle zu vervollständigen, werden vom Anwender die gewählten Anforderungen an das Testwerkzeug und die gewählten Bewertungskriterien zu den Testfällen hinzugefügt.

**Ausgangsschnittstelle:** Die Ausgangsschnittstelle übergibt konkrete Testfälle, wie sie in Kapitel 2.2 definiert wurden. Die Testfälle beinhalten das logische Szenario, die konkreten Szenarioparameter, die Anforderungen an das Testwerkzeug und die Bewertungskriterien.

### Modul: Testdurchführung

**Kurzbeschreibung:** Die Testdurchführung besteht aus den Modulen der Testfallzuordnung und Testtreiber, welche in Kapitel 9 im Detail beschrieben werden. In der Testfallzuordnung werden die einzelnen Testfälle anhand einer Zuordnungsmethode auf die verfügbaren X-in-the-Loop-Verfahren verteilt. Die Eingangs- und Ausgangsdaten des Testobjekts werden durch verschiedene Testtreiber zur Verfügung gestellt und verarbeitet.

**Eingangsschnittstellen:** Eingangsschnittstellen für dieses Modul sind die Testfälle inklusive der Anforderungen an die Testwerkzeuge. Eine weitere Eingangsschnittstelle ist die Information über verfügbare X-in-the-Loop-Verfahren sowie Simulationsmodelle und deren Güte.

**Ausgangsschnittstellen:** Die Ausgangsschnittstelle der Testdurchführung übergibt die messbaren und beobachtbaren Ausgangsdaten des Testobjekts an die Testfallauswertung.

### Modul: Testfallauswertung

**Kurzbeschreibung:** Das Modul der Testfallauswertung beinhaltet die Bewertung der Testergebnisse anhand von verschiedenen Bewertungsfunktionen und die Analyse der Testergebnisse auf Parameterwirkungen. Das Modul wird in Kapitel 10 detailliert vorgestellt.

**Eingangsschnittstellen:** Die Eingangsschnittstelle der Bewertung und Analyse sind die vom Anwender gewählten Bewertungskriterien und die messbaren beziehungsweise beobachtbaren Ausgangsdaten des Testobjekts.

**Ausgangsschnittstellen:** Die Ausgangsschnittstelle übergibt Berichte, die Parametervariationen aufzeigen, bei denen das System eine „hohe“ oder „geringe“ Performance beziehungsweise Güte bezüglich der gewählten Bewertungskriterien zeigt. Diese Ergebnisse können in einer weiteren Iteration des Testbaukastens im Modul der Auswahl und Analyse verwendet werden, um Parameterbereiche für weitere Tests zu identifizieren.



## 7 Auswahl und Analyse von Einflussparametern

Dieses Kapitel beschreibt das Modul der Auswahl und Analyse von Einflussparametern des modularen virtuellen Testbaukastens. Die eigentliche Aufgabe ist die systematische Ableitung von Einflussparametern und Diskretisierungsstufen für die Testfallerstellung. In diesem Kapitel werden dazu Informationsquellen für die Auswahl und Analyse vorgestellt. Weiterhin wird die Herausforderung der Diskretisierung von wertkontinuierlichen Parametern thematisiert. Eine Methode für die Diskretisierung wird jedoch nicht vorgestellt.

Zunächst werden Anforderungen an die Auswahl und Analyse definiert. Daraufgehend wird der relevante Stand der Technik zur Auswahl und Analyse von Einflussparametern in Testkonzepten beschrieben und mit den gestellten Anforderungen verglichen. Auf dieser Basis wird der Forschungsbedarf abgeleitet. Anschließend werden Informationsquellen für die Auswahl und Analyse von Einflussparametern analysiert und anhand der gestellten Anforderungen bewertet. Abschließend wird ein Schema zur Beschreibung von Einflussparametern vorgestellt und die exemplarische Anwendung des Moduls an der in Kapitel 3.3 vorgestellten Fahrfunktion gezeigt.

Die in dieser Arbeit betrachteten Einflussparameter beschreiben die Szenarien und die beinhalteten Szenen, in denen das Testobjekt getestet wird. Somit findet eine Betrachtung von außen auf das Testobjekt statt. Eine Analyse vom Inneren des Systems findet bei diesem Ansatz derzeit nicht statt. Einen Ansatz zur Identifikation von Einflussparametern durch die Analyse des inneren Aufbaus des Systems liefert beispielsweise Reschka (2016a, Kap.8). Eine Erweiterung des Testbaukastens beziehungsweise eine Kombination mit den von Reschka (2016a, Kap.8) vorgestellten Ansätzen ist in weiteren Forschungsarbeiten denkbar, wird jedoch in der vorliegenden Arbeit nicht weiter verfolgt.

### 7.1 Anforderungen

Im Folgenden werden die Anforderungen an die Auswahl und Analyse der Einflussparameter beschrieben. Die Auswahl wird durchgeführt, um Parameter zur Beschreibung des Szenarios zu identifizieren, die einen Einfluss auf das Testobjekt haben. Dabei müssen zugleich gültige und ungültige Wertebereiche identifiziert werden. Darauf aufbauend wird die Analyse verwendet, um für die identifizierten Wertebereiche Diskretisierungsstufen abzuleiten. Die Anforderungen basieren auf den Erfahrungen des Autors, die im Projekt des Engstellenassistenten gesammelt wurden. Weiterhin werden die Anforderungen an ein Schema für die Darstellung von Einflussparametern definiert, um die erfassten Parameter und Diskretisierungsstufen an die Testfallerstellung zu übergeben. Neben den Anforderungen, die in diesem Kapitel an die Auswahl und Analyse gestellt werden, gelten die in Kapitel 5 definierten Anforderungen bezüglich der Effizienz und Effektivität.

### 7.1.1 Anforderungen an die Auswahl von Einflussparametern

**Die Auswahl muss Einflussparameter identifizieren, die einen Einfluss auf das Testobjekt haben.**

Das Teilmodul der Auswahl von Einflussparametern wird verwendet, um Parameter für die Beschreibung von Szenarien und Testfällen, bezogen auf das Testobjekt, zu identifizieren. Dabei müssen Parameter identifiziert werden, die einen direkten oder indirekten Einfluss von außen auf das Testobjekt haben. Die Parameter sollen im 4-Ebenen-Modell, welches in Kapitel 8.3 vorgestellt wird, zur Beschreibung von Szenarien hinterlegt werden.

**Die Auswahl muss repräsentative Wertebereiche identifizieren.**

Neben der Identifikation der Einflussparameter müssen repräsentative Wertebereiche, bezogen auf die jeweiligen Testobjekte, bestimmt werden. Der Begriff der Repräsentativität orientiert sich an der Definition der Statistik. Dort ist eine Stichprobe repräsentativ, wenn „sie die innere Struktur und die Charakteristika einer statistischen Grundgesamtheit adäquat abbildet“ (Eckstein, 2014, S.16). Damit sind Wertebereiche in der vorliegenden Arbeit repräsentativ, wenn diese die Zielumgebungen und Anwendungsszenarien des Testobjekts adäquat abbilden. Daher ist neben den maximal und minimal möglichen Werten auch eine statistische Verteilung innerhalb der Wertebereiche abzubilden. Die repräsentativen Wertebereiche bieten das Potential, in der Testfallerstellung repräsentative Testfälle abzuleiten.

### 7.1.2 Anforderungen an die Analyse von Einflussparametern

**Die Analyse muss Diskretisierungsstufen für wertkontinuierliche Parameter bestimmen.**

Die Analyse muss für die Wertebereiche der identifizierten Einflussparameter Diskretisierungsfunktionen bestimmen. Die Diskretisierungsfunktion beschreibt die Verteilung der Diskretisierungsstufen über die identifizierten Wertebereiche der jeweiligen Parameter. Die Funktionen sind so zu wählen, dass Parameterstufen umgesetzt werden, die fehlersensitiv sind oder einer statistischen Verteilung für die Zielumgebungen entsprechen. Parameterstufen haben eine hohe Fehlersensitivität, wenn bei diesen Stufen häufig Fehler auftreten oder die Stufen für das Testobjekt herausfordernd sind. Dies können nach Liggesmeyer (2009, S.196) die Grenzwerte der identifizierten Wertebereiche sein. Eine statistische Verteilung von Parameterwerten für die Zielumgebungen kann, wie von Prialé Olivares u. a. (2016) gezeigt, für die Ableitung von Diskretisierungsstufen verwendet werden. Allerdings stellt die Identifikation von pathologischen Fällen<sup>1</sup> bei der statistischen Auswertung eine Herausforderung dar. Pathologische Fälle sind nach Reschka u. a. (2015b, S.919) die Fälle, die nur mit einer sehr geringen Wahrscheinlichkeit auftreten. Daher sollte für eine statistische Auswertung gezeigt werden, dass eine Menge von repräsentativen Stichproben erfasst wurde, um auch pathologische Fälle zu erfassen. Weiterhin ist die Sensitivität der Fahrfunktion bei der Wahl der Diskretisierungsstufen, wenn möglich, zu beachten. In einem Wertebereich, in dem die

<sup>1</sup>Pathologische Fälle werden im Englischen mit *corner cases* oder *edge cases* übersetzt. Diese Begriffe werden wiederum ins Deutsche mit dem Begriff der Grenzfälle übersetzt. Jedoch sind pathologische Fälle nicht mit Grenzfällen gleichzusetzen. Ein Grenzfall führt die Fahrfunktion an die Grenze ihrer Funktionalität. Dies kann eintreten, wenn ein Eingangsparameter, beispielsweise die Relativgeschwindigkeit, einen „sehr hohen“ Wert einnimmt. Ein pathologischer Fall ist hingegen ein Fall, der nur mit einer sehr geringen Wahrscheinlichkeit auftritt. Dies kann, muss aber nicht, ein Grenzfall sein. Nicht jeder Grenzfall ist ein pathologischer Fall und andersherum.

Fahrfunktion eine erhöhte Sensitivität zeigt, sollte eine höhere Auflösung der Diskretisierung gewählt werden, um ein genaueres Testen zu ermöglichen. Dies setzt wiederum ein gutes Systemverständnis voraus.

**Die Analyse muss die Fehlerwahrscheinlichkeit durch die Diskretisierung von wertkontinuierlichen Parametern bestimmen.**

Durch die Diskretisierung der wertkontinuierlichen Einflussparameter wird keine vollständige Testabdeckung erreicht, da zwischen zwei Diskretisierungsstufen theoretisch unendlich viele weitere Stufen vorhanden sind, die nicht getestet werden (Kaiser, 2015b). Daher wird mittels der durchgeführten Testfälle nur ein bestimmter Anteil von Fehlern aufgedeckt, wodurch weiterhin eine Fehlerwahrscheinlichkeit für das Testobjekt vorhanden ist. Somit soll die Analyse der identifizierten Einflussparameter eine Aussage über die Fehlerwahrscheinlichkeit des nicht getesteten Bereichs treffen.

### 7.1.3 Anforderungen an ein Schema zur Darstellung von Einflussparametern

**Das Schema muss verschiedene Wertebereiche und Diskretisierungsfunktionen für wertkontinuierliche Einflussparameter beschreiben können.**

Das Schema muss für die Einflussparameter getrennte Wertebereiche abbilden können. Die Wertebereiche dürfen sich dabei nicht widersprechen beziehungsweise gegenseitig ausschließen. Weiterhin wird die Forderung gestellt, dass verschiedene Funktionen zur Diskretisierung von wertkontinuierlichen Parametern abgelegt werden können. Damit ist es möglich, die verschiedenen Wertebereiche in unterschiedlich große Diskretisierungsstufen zu unterteilen.

**Das Schema muss die Betrachtung verschiedenster Arten von Informationsquellen ermöglichen.**

Für die Auswahl und Analyse von Einflussparametern stehen verschiedenste Informationsquellen zur Verfügung. Um die Informationen dieser Quellen nutzen zu können, muss das Schema die Informationen von verschiedensten Informationsquellen aufnehmen können.

**Das Schema muss die Beschreibung der Parameter in einer formalisierbaren Darstellung ermöglichen.**

Die Ergebnisse der Auswahl und Analyse müssen in einer formalisierbaren Darstellung beschreibbar sein. Dies ist notwendig, damit das Teilmodul der Auswahl die identifizierten Parameter an das Teilmodul der Analyse übergeben kann. Weiterhin wird eine formalisierbare Darstellung benötigt, um die Ergebnisse der Analyse an das Modul der Testfallerstellung zu übergeben.

## 7.2 Stand der Technik

Eine Literaturrecherche zur Auswahl und Analyse zeigt, dass die eigentlichen Herausforderungen einer systematischen Auswahl und Analyse von Einflussparametern bisher noch wenig thematisiert werden. Derzeit findet die Auswahl der Parameter und Diskretisierungsstufen meist auf der Basis von Expertenwissen statt. Das Forschungsprojekt *PEGASUS*<sup>2</sup> zeigt

---

<sup>2</sup>Projekt zur Etablierung von generell akzeptierten Gütekriterien, Werkzeugen und Methoden sowie Szenarien und Situationen zur Freigabe hochautomatisierter Fahrfunktionen

durch die Wahl der Forschungsschwerpunkte die gleichen Erkenntnisse (Plättner u. a., 2016). Daher beschreibt das folgende Kapitel einen ausgewählten Teil zum Stand der Technik aus unterschiedlichen Testkonzepten, die vor ähnlichen Herausforderungen stehen wie der modulare virtuelle Testbaukasten. Die Auswahl bezieht sich auf Ansätze, die Systeme im Automobilkontext testen und im Ansatz eine Auswahl und Analyse von Einflussparametern direkt oder indirekt vornehmen. Die verschiedenen Ansätze werden vorgestellt und mit den aufgestellten Anforderungen verglichen. Alle analysierten Ansätze betrachten das Testobjekt, wie in der vorliegenden Arbeit, als Black-Box.

Eltaher (2013) stellt in seiner Arbeit einen kognitivorientierten Testansatz vor. Die zentrale Idee ist das Erlernen der Testmethoden von Testexperten, um die Methoden auf ein Expertentestsystem zu übertragen. Nach Form und Eltaher (2011) bieten Expertentests eine Ergänzung zum systematischen Test, da Experten zum Teil noch Fehler nach einem systematischen Test im System finden. In der Arbeit wird herausgestellt, dass das Wissen über die Testmethoden nur in implizierter Form vorliegt und damit nicht direkt für einen systematischen Ansatz verwendbar ist. Anhand ihrer Auswahl von Testfällen führen die Experten eine implizite Auswahl und Analyse der Einflussparameter durch, wodurch der von Eltaher (2013) vorgestellte Ansatz zum Stand der Technik gehört. Eltaher (2013) verwendet für seinen Ansatz *Situations-Operator-Modelle*, um das implizierte Wissen der Experten für ein Expertensystem zu modellieren. In einer Fallstudie mit einem Infotainmentsystem wird demonstriert, dass ein von Experten trainiertes System, welches mit dem vorgestellten Ansatz trainiert wurde, in Relation zu einem durchschnittlichen menschlichen Tester weitere Fehler im System findet, die vom durchschnittlichen Tester übersehen wurden. Das Infotainmentsystem ist ein Testobjekt mit diskreten Eingangsdaten, wodurch eine Diskretisierung von wertkontinuierlichen Eingangsparametern nicht notwendig ist. Daher steht der Wertebereich und die Diskretisierung der Wertebereiche bereits fest. Eine Auswahl der Einflussparameter durch die Testexperten wurde im Ansatz nicht betrachtet. Dies gilt ebenso für die Analyse der Einflussparameter, da die Diskretisierung der Eingangsparameter bereits vorgegeben war. Die Fehlerwahrscheinlichkeit könnte bei diesem eingeschränkten System durch einen erschöpfenden Test, bei dem alle möglichen Kombinationen von Diskretisierungsstufen getestet werden, auf 0% reduziert werden. Die Skalierbarkeit des Ansatzes auf aktuelle Infotainmentsystemen wird nicht gezeigt. Daher kann keine Aussage darüber getroffen werden, ob auch bei aktuellen Infotainmentsystemen eine Fehlerwahrscheinlichkeit von 0% erreicht werden kann. Wie der Ansatz mit wertkontinuierlichen Parametern verwendet werden kann und wie in diesem Fall die Fehlerwahrscheinlichkeit bestimmbar ist, wird nicht gezeigt.

Lindlar (2012) und Kruse u. a. (2009) präsentieren modellbasierte evolutionäre Testmethoden. Kruse u. a. (2009) stellen den Ansatz anhand eines Antiblockiersystem vor. Lindlar (2012) demonstriert seinen Ansatz anhand eines Abstandsregeltempomaten. Die Eingangsparameter für den Abstandsregeltempomaten sind die Geschwindigkeit des Fahrzeugs vor dem Ego-Fahrzeug, die Fahrereingaben zum System und die Krümmung der Straße (Lindlar, 2012, S.102f). Eine Herleitung der Auswahl der Parameter sowie der Wertebereiche findet nicht statt. Ein evolutionärer Algorithmus optimiert das Testobjekt mittels der Durchführung von 100.000 Testfällen (10 Optimierungsdurchläufe mit 100 Generationen mit jeweils 100 Populationen) (Lindlar, 2012, S.111). Die Testergebnisse werden über eine Fitnessfunktion bewertet, die die Testergebnisse auf einen atomaren Wert zurückführen. Durch die Optimierung werden die Testfälle identifiziert, bei denen das System den geringsten Fitnesswert erreicht. Ein Vor-

teil dieses Ansatzes ist die nicht benötigte vorherige Diskretisierung der wertkontinuierlichen Parameter, da der evolutionäre Algorithmus die Diskretisierungsstufen für den nächsten Test iterativ berechnet. Anhand des vorgestellten Ansatzes können keine Aussagen über die verbleibende Fehlerwahrscheinlichkeit getroffen werden. Aufgrund der begrenzten Anzahl von Populationen und Optimierungsdurchläufen kann nicht gezeigt werden, dass innerhalb der Optimierungsschritte alle Extrema gefunden wurden. Eine weitere Herausforderung bei der Anwendung dieser Testmethode ist die Definition einer Fitnessfunktion, die das Ergebnis eines Testfalls auf einen atomaren Zahlenwert reduziert, da gegebenenfalls konkurrierende Bewertungskriterien gleichzeitig betrachtet werden müssen.

Junghanns u. a. (2008) sowie Tatar und Mauss (2014) stellen das Testwerkzeug *TestWeaver* vor, welches eine Alternative zum evolutionären Testen darstellt. Es handelt sich dabei um eine Software zur automatisierten Testfallerstellung. Ziel ist das Erreichen von Systemzuständen, bei denen das System seine Spezifikation verletzt. Dies soll durch die Erstellung von konkreten Testfällen aus den Systemantworten der vorherigen Testfälle erreicht werden. Die Autoren vergleichen die Strategie mit einem Schachspiel gegen das System. Als Eingangsdaten werden diskrete Wertebereiche benötigt, die von Junghanns u. a. (2008) Partitionen genannt werden. Da das Tool bereits diskrete Wertebereiche für die Testfallerstellung benötigt, wird die Auswahl und Analyse von Einflussparametern nicht weiter betrachtet und thematisiert. Hilf u. a. (2010) zeigen die Anwendung des Testwerkzeugs anhand einer Seitenwindstabilisierungsfunktion. Als Eingangsdaten werden die Straße, Windeigenschaften, Gas- und Bremspedalstellung sowie der Lenkwinkel des Fahrzeugs benannt. Informationsquellen für die Auswahl und Diskretisierung der Eingangsparameter werden nicht diskutiert. Jedoch zeigt der Ansatz, wie aus existierenden Testergebnissen neue Testfälle erstellt werden können.

### **Bewertung der Ansätze und Ableitung des Forschungsbedarfs**

Die ausgewählte Literatur zeigt, dass die Auswahl und Analyse von Einflussparametern derzeit in den vorgestellten Testkonzepten eine geringe Betrachtung findet und die Testkonzepte vor ähnlichen Herausforderungen stehen wie der modulare virtuelle Testbaukasten. Die Auswahl und Analyse der Einflussparameter basiert auf Expertenwissen oder für die Anwendung der vorgestellten Ansätze werden bereits diskretisierte Eingangsparameter benötigt. Wie eine Diskretisierung durchführbar ist, wird nicht diskutiert. Kacker u. a. (2013) treffen ebenfalls die Aussage, dass die Definition von Eingangsdaten und damit auch die Diskretisierung von wertkontinuierlichen Parametern ein Forschungsthema darstellt. Derzeit können Expertentests und das evolutionäre Testen Ansätze für eine Analyse zur Erstellung von Diskretisierungsstufen liefern. Expertentests können jedoch nicht für einen systematischen Test verwendet werden, da das Wissen lediglich in impliziter Form vorliegt. Für die Anwendung der evolutionären Algorithmen ist eine große Anzahl von Testfällen notwendig, die bedingt durch die verfügbaren Testdurchführungen nicht immer ökonomisch durchgeführt werden können. Daher müssen weitere Informationsquellen für eine systematische Auswahl und Analyse von Einflussparametern bewertet werden. Eine solche Übersicht über mögliche Informationsquellen sowie eine Betrachtung, welche Anforderungen von diesen Informationsquellen erfüllt werden, wurde bisher nicht erstellt. Daher wird im folgenden Kapitel eine Übersicht über Informationsquellen für die Auswahl und Analyse von Einflussparametern entwickelt und die Erfüllung der Anforderungen durch die vorgestellten Informationsquellen diskutiert.



## 7.3 Informationsquellen zur Auswahl von Einflussparametern

### Ableitung durch die Funktionsspezifikation und darin enthaltende Anwendungsszenarien

Die Funktionsspezifikation definiert das gewünschte Verhalten in den Anwendungsszenarien des Systems. Dadurch werden viele Bestandteile der Szenen beschrieben, die für das System von Bedeutung sind. Beispielsweise kann die Zielumgebung einer Fahrfunktion als Autobahn definiert werden. Damit wird die Auswahl von Einflussparametern auf die Elemente von Szenen priorisiert, die für eine Autobahnszene besonders charakteristisch sind. Daher lassen sich aus der Spezifikation logische Szenarien mit einer Beschreibung der Szenen und deren Elementen ableiten. Zudem lässt sich die Existenz von bestimmten Elementen in den Szenen und deren grundsätzlichen Eigenschaften herleiten.

Eine Herausforderung bei der Verwendung von Funktionsspezifikationen ergibt sich, wenn die Spezifikation für die Anwendungsfälle nicht vollständig oder nicht korrekt beschrieben wurde und das gewünschte Verhalten eventuell falsch interpretiert wird (Leveson, 2011, S.47ff). Nach Fleischmann (2008, S.13ff) sind je nach Studie 40% bis 60% aller schweren Fehler im Automobilbereich auf Fehler in der Anforderungsanalyse zurückzuführen. Reschka (2016a, Kap.4,5) kritisiert, dass es keinen genormten Prozess zur Ableitung von Spezifikationen gibt und dadurch die Spezifikation unvollständig sein kann. Dies kann besonders bei Fahrfunktionen auftreten, die nur prototypisch für die Forschung umgesetzt werden. Dort ist die Funktionsbeschreibung zu Beginn des Projekts meist nur vage bekannt und verändert sich im Laufe des Forschungsprojekts. Die Herausforderung für die Auswahl von Einflussparametern besteht darin, dass durch fehlerhafte, unvollständige oder nicht sorgfältig erarbeitete Spezifikationen ebenfalls die Auswahl der Einflussparameter fehlerhaft oder unvollständig sein kann. Dadurch können falsche Wertebereiche getestet werden, die in den Anwendungsszenarien nicht auftreten. Damit werden durch eine Ableitung von Testfällen auf Basis der Spezifikation, die im Folgenden in den realen Anwendungsszenarien durchgeführt werden, Fehler bei der Definition von Parameterwerten der Anwendungsszenarien identifiziert. Daraus ergibt sich die Konsequenz, dass die Definition der Szenarien und die Auswahl der Parameter eigentlich bereits bei der Erstellung der Spezifikation durchgeführt werden muss, um die Parameter für die jeweiligen Anwendungsszenarien abzuleiten. Dies bedeutet, dass eine sorgfältig erarbeitete Spezifikation benötigt wird, um diese Informationsquelle für die Auswahl von Einflussparametern zu nutzen (Reschka, 2016b).

### 4-Ebenen-Modell zur Beschreibung von Szenarien

Das 4-Ebenen-Modell zur generischen Beschreibung von Szenarien, welches in Kapitel 8.3 vorgestellt wird, ist eine weitere Informationsquelle für die Auswahl von Elementen und deren Eigenschaften in der Szene. Das Modell wird für eine systematische Beschreibung und Darstellung der Umgebung der Fahrfunktion verwendet. Durch die mehrfache Verwendung des Modells beim Test von verschiedensten Fahrfunktionen wird dies sukzessive erweitert und kann so als Informationsspeicher für identifizierte Elemente und deren Eigenschaften verwendet werden.

Das 4-Ebenen-Modell wird jedoch nie die Realität vollständig repräsentieren, da durch die Modellbildung und Simulation immer Vereinfachungen getroffen werden, die von der Realität

abweichen. Somit kann das 4-Ebenen-Modell eine Stütze für die Auswahl von Parametern darstellen, jedoch sollten weitere Informationsquellen hinzugezogen werden.

### **Normen und Richtlinien zur Anlage von Straßen**

Aus den Normen und Richtlinien zur Anlage von Straßen verschiedener Typen, wie Autobahnen (FGSV, 2008), Landstraßen (FGSV, 2013) und Stadtstraßen (FGSV, 2006), können Eigenschaften über stationäre Elemente der Straße abgeleitet werden. Zusätzlich können Wertebereiche der Eigenschaften, wie die Größe der Elemente oder mögliche Abstände, bestimmt werden. Dadurch lassen sich die minimal oder maximal erlaubten Grenzwerte für die Wertebereiche der Eigenschaften von stationären Elementen ableiten.

Eine Aussage über die Eigenschaften und Wertebereiche von dynamischen Elementen können aus den Normen und Richtlinien nicht direkt abgeleitet werden. Einige Werte, wie die maximale Geschwindigkeit von dynamischen Verkehrsteilnehmern, können durch die Normen und Richtlinien geschätzt werden. Jedoch besteht nicht die Möglichkeit, auf Basis der Normen und Richtlinien eine Aussage darüber zu treffen, ob die Verkehrsteilnehmer sich an die Normen und Richtlinien halten und beispielsweise die zulässige maximal erlaubte Geschwindigkeit einhalten. Dies wird umso deutlicher bei der Betrachtung von pathologischen Fällen, die durch das Verhalten von dynamischen Verkehrsteilnehmern hervorgerufen werden, die sich nicht regelkonform verhalten beziehungsweise für deren Verhalten keine Normen und Richtlinien existieren. Daraus folgt, dass die Normen und Richtlinien lediglich für die Auswahl der Eigenschaften von stationären Elementen geeignet sind.

Weiterhin müssen für die Verwendung dieser Informationsquelle die Normen und Richtlinien für die Zielumgebung des Testobjekts existieren. In Ländern mit einem strukturierten Verkehr und existierenden Normen und Richtlinien für die Anlage von Straßen, wie Deutschland (FGSV, 2008), USA (NRC, 2010) oder Australien (TRM, 2013), können die Normen und Richtlinien zur Beschreibung der Eigenschaften und Wertebereiche der stationären Straßeninfrastruktur verwendet werden. In Ländern, in denen keine Normen und Richtlinien existieren, können diese nicht als Informationsquelle verwendet werden. Dies stellt ein Nachteil dieser Informationsquelle dar.

Ein weiterer Nachteil dieser Informationsquelle ist das mögliche Delta zwischen normgerechten und existierenden Straßen. Durch die Bestimmung der Grenzen der Wertebereiche aus den Normen und Richtlinien werden Straßen im normgerechten Wertebereich getestet. Alle Straßen, die nicht nach den Normen und Richtlinien gebaut wurden, werden nicht getestet, wodurch sich das Testobjekt bei einem Betrieb auf diesen nicht normgerechten Straßen im ungetesteten Bereich befindet. Daher darf die Fahrfunktion bei der alleinigen Verwendung dieser Informationsquelle, nur auf Straßen eingesetzt werden, die nach Normen und Richtlinien gebaut sind.

### **Ableitung anhand von Fahrmanövern**

Eine weitere Informationsquelle für die Auswahl von Parametern für die Elemente der Szenen ist die Ableitung von Größen aus Fahrmanövern. Die Definition eines Fahrmanövers wurde in Kapitel 2.3 gegeben. Tölle (1996, S.32ff) definiert neun Fahrmanöver in einer Basismenge von Manövern, die ein automatisiertes Fahrzeug beherrschen muss, um am öffentlichen Straßenverkehr teilzunehmen. Die von Tölle genannten Fahrmanöver sind *Anfahren*, *Folgen*, *Annähern*, *Passieren*, *Kreuzung überqueren*, *Fahrstreifenwechsel*, *Abbiegen*, *Umkehren* und *Parken*.



Da diese Fahrmanöver von einem automatisierten Fahrzeug beherrscht werden müssen, um am Straßenverkehr teilzunehmen, lassen sich daraus direkt die Fahrmanöver für die Fahrzeuge ableiten, die sich um das Testobjekt herumbewegen. Die Fremdfahrzeuge müssen diese Fahrmanöver ebenfalls beherrschen, um am Straßenverkehr teilzunehmen. Daher lassen sich Eigenschaften, wie Geschwindigkeit, Beschleunigung oder Lenkwinkel für dynamische Elemente im Szenario aus dieser Informationsquelle auswählen.

Wertebereiche können jedoch nur bedingt abgeleitet werden, da diese direkt von den Verkehrsteilnehmern in der Szene und dem individuellen Verhalten der Fahrer in den Fahrzeugen abhängig sind. Eine Möglichkeit für die Bestimmung der Wertebereiche für Fahrmanöver anderer Verkehrsteilnehmer ist die Schätzung aus Messdaten, wie von Ernst (2015) vorgestellt. Einen weiteren starken Einfluss hat das regionale Fahrverhalten der Verkehrsteilnehmer. Je nach Region verhält sich der Verkehr unterschiedlich, wie der Vergleich zwischen schwedischem und chinesischem Verkehr von Lindgren u. a. (2008) zeigt.

### Fahrzeugkataloge

Fahrzeugkataloge können verwendet werden, um Parameter von beweglichen Elementen zu identifizieren. Diese Parameter sind unter anderem die technischen Daten, wie die geometrischen Maße oder die Leistung von verschiedenen Fahrzeugklassen. Damit besteht die Möglichkeit, die Parameter und Wertebereiche für die beweglichen Fahrzeuge zu bestimmen und somit Größen für die theoretisch mögliche Dynamik der Elemente abzuleiten. Die Informationsquelle der Fahrzeugkataloge stellt eine Analogie zu den Normen und Richtlinien dar. Während durch die Normen und Richtlinien die theoretisch möglichen Wertebereiche für stationäre Elemente identifizierbar sind, können durch die Fahrzeugkataloge die möglichen Wertebereiche für bewegliche Fahrzeuge bestimmt werden.

### Messfahrten im öffentlichen Straßenverkehr

Eine weitere Möglichkeit Einflussparameter auszuwählen, ist die Durchführung von Messfahrten auf existierenden Straßen. Damit lassen sich sowohl Wertebereiche als auch statistische Verteilungen von Diskretisierungsstufen von stationären und beweglichen Elementen identifizieren. Ein Beispiel in Form von Flottenversuchen ist das von Benmimoun u. a. (2012a) vorgestellte *euroFOT*-Projekt. Zofka u. a. (2015) zeigen in einem Ansatz, wie aus Messfahrten die Wertebereiche für Fahrstreifenbreiten durch Partikelfilter geschätzt werden. Winner (2002) stellt den Ansatz des *Trojanischen Pferdes* vor. Beim Ansatz werden der zu testenden Fahrfunktion im Fahrzeug alle benötigten Eingangssignale zur Verfügung gestellt. Die Ausgangssignale werden jedoch nicht an die Aktorik übergeben. Zur Bewertung des Systems werden die Ausgangssignale mit den Signalen eines Referenzsystems, wie zum Beispiel ein menschlicher Fahrer, verglichen. Tesla zeigt einen ähnlichen Ansatz, um die Funktion ihres *Autopiloten* durch Fahrten von Kunden kontinuierlich zu verbessern<sup>3</sup>. Die Messfahrten werden ebenfalls für die Erstellung von Testfällen verwendet<sup>4</sup>. Die Ansätze von Tesla und Winner (2002) zeigen wie Massendaten (Frei übersetzt: big data) für die Verbesserung von Fahrfunktionen verwendet werden können.

<sup>3</sup><http://electrek.co/2015/10/30/the-autopilot-is-learning-fast-model-s-owners-are-already-reporting-that-teslas-autopilot-is-self-improving/>, abgerufen: 28.02.2016

<sup>4</sup><https://www.technologyreview.com/s/601567/tesla-tests-self-driving-functions-with-secret-updates-to-its-customers-cars>, abgerufen: 26.05.2016

Der Einsatz und die Auswertung von Messfahrten ist mit einem hohen Aufwand verbunden. Die Messfahrten können zum einen nur in Realzeit durchgeführt werden und müssen zum anderen anschließend ausgewertet werden (Benmimoun u. a., 2012c). Die Auswertung geschieht in aufwändigen manuellen Beschriftungsprozessen (engl. labeling), wie bei Kamerabildern, oder durch manuelle bis halbautomatische Verfahren zur Auswertung, um bestimmte Szenen oder Situationen in den Messfahrten wiederzuerkennen. Ansätze zur automatischen Erkennung von Situationen liefern zum Beispiel Benmimoun u. a. (2010).

Damit besteht die Möglichkeit, mit Hilfe von Messfahrten Wertebereiche zu erstellen. Jedoch erfasst eine Durchführung von Messfahrten immer nur eine Stichprobe von repräsentativen Szenen. Somit besteht immer eine Wahrscheinlichkeit, dass nicht alle repräsentativen Szenen erfasst werden. Dadurch hängt der Ansatz direkt von der Gestaltung der Stichprobe ab.

### **Ableitung von Verkehrsszenarien aus Unfallanalysen**

Die Unfallanalysen vom statistischen Bundesamt (Statistisches Bundesamt, 2016) stellen eine weitere Informationsquelle für die Auswahl von Parametern dar. Durch die Analyse von Unfällen können Szenarien abgeleitet werden, bei denen das Fehlverhalten von Menschen zu Unfällen geführt hat. Durch die Analyse lassen sich komplette Szenarien mit einer Abfolge von Szenen mit der Szenerie und dynamischen Elementen ableiten. Dies ist gegenüber den bisher vorgestellten Informationsquellen ein Vorteil, da neben konkreten Eigenschaften auch Wertebereiche und Diskretisierungsstufen für die Einflussparameter aus den Unfällen abgeleitet werden können.

Jedoch lassen sich durch die Analyse der Unfallzahlen nur Szenarien analysieren, die für den Menschen als Fahrer kritisch waren und zum Unfall geführt haben. Derzeit kann keine Aussage darüber getroffen werden, ob für das Testobjekt beziehungsweise die automatisierte Fahrfunktion das selbe Szenario kritisch ist. Gegebenenfalls sind für das Testobjekt andere Szenarien kritisch, bei denen das Testobjekt einen Unfall verursachen würde (Gasser u. a., 2012, S.11). Ein weiterer Kritikpunkt an dieser Informationsquelle ist die Tatsache, dass nur Unfälle analysiert werden können. Beinaheunfälle werden in der Statistik nicht erfasst und stehen damit durch die Informationsquelle nicht zur Verfügung. Um an diese Daten zu gelangen, müssten andere Informationsquellen, wie Messfahrten, verwendet werden.

### **Expertenwissen**

Eine weitere Informationsquelle für die Auswahl von Einflussparametern ist das Expertenwissen. Experten können auf Basis ihrer Expertise Szenarien inklusive Szenen und beinhaltende Elementen identifizieren, bei denen Fehler besonders oft auftreten. Des Weiteren bestimmen die Experten fehlersensitive Diskretisierungsstufen durch ihre langjährige Erfahrung mit den Systemen mittels der Auswahl von konkreten Testfällen. Dabei ist zu beachten, dass das Expertenwissen jedoch nur bei einer evolutionären Entwicklung verfügbar ist. Bei einer disruptiven Entwicklung ist kein Expertenwissen verfügbar. Die Testmethode ist auch unter dem Begriff des *Error Guessing* bekannt (Homès, 2012, S.190).

Nach Eltaher (2013, S.4) kann das Wissen von den Experten meist nur implizit beschrieben werden, da die Experten bei Testprozessen oftmals intuitiv handeln. Dabei werden von den Experten beim Testen des Systems intuitiv Elemente im Szenario mit Diskretisierungsstufen getestet, indem beispielsweise ein bestimmter Streckenabschnitt gefahren wird, bei dem die Experten wissen, dass das zu testende System oft Fehler zeigt. Eltaher (2013) beschreibt eine Methode, um das Wissen und die Teststrategie von Experten, die ein Infotainmentsystem

testen, zu objektivieren und auf ein Expertensystem zu übertragen. Bisher wurde jedoch nicht gezeigt, dass der von Eltaher (2013) vorgestellte Ansatz für den Test von automatisierten Fahrfunktionen verwendbar ist. Falls dies der Fall wäre, könnte das objektivierte Wissen zur Auswahl von Einflussparametern verwendet werden. Damit kann ein Teil der pathologischen Fälle, d.h. Fälle die sehr selten auftreten, durch das Wissen der Experten systematisch abgeleitet werden.

## 7.4 Informationsquellen zur Analyse von Einflussparametern

Nach der Auswahl und Identifikation der Einflussparameter müssen diese anschließend analysiert werden, um in der Testfallerstellung verwendet zu werden. Im Folgenden werden ausgewählte Informationsquellen und Herangehensweisen zur Analyse von Einflussparametern beschrieben und bewertet.

### Optimierungsalgorithmen

Optimierungsalgorithmen bieten eine Möglichkeit zur Analyse von Einflussparametern. Basierend auf Zielfunktionen variieren die Optimierungsalgorithmen die Eingangsdaten, sodass die Zielfunktionen minimiert oder maximiert werden. Ein Vorteil der Anwendung von Optimierungsalgorithmen ist die nicht notwendige vorherige Diskretisierung von wertkontinuierlichen Parametern, da die Diskretisierungsstufen implizit durch den Optimierungsalgorithmus berechnet werden. Damit können Diskretisierungsstufen der Wertebereiche identifiziert werden, bei denen das Testobjekt, bewertet nach den Zielfunktionen, die beste oder schlechteste Performance zeigt.

Voraussetzung für die Verwendung von Optimierungsalgorithmen, wie beim evolutionären Testen, ist eine geschlossene Testschleife über Testfallerstellung, Testdurchführung und Testfallauswertung. Des Weiteren muss für eine ökonomische Nutzung der Algorithmen eine Testdurchführung schneller als in Echtzeit oder eine hohe Parallelisierung der Testdurchführung möglich sein, da eine große Anzahl von Tests für die Optimierung notwendig ist (Kaiser, 2015a). Lindlar (2012) hat in seiner Arbeit die Optimierung der Tests eines Abstandsregeltempomaten in 10 Optimierungsschritten mit 100 Generation und jeweils 100 Populationen pro Generation durchgeführt. Damit wurden 100.000 Testfälle für die Optimierung durchgeführt und ausgewertet. Kruse u. a. (2009) nutzen für den Test eines Antiblockiersystems ebenfalls Optimierungsalgorithmen. In diesem Beispiel werden 50.000 Testfälle durchgeführt.

Ein Nachteil dieses Ansatzes ist die Definition einer Zielfunktion, die das Ergebnis jedes Testfalls einem „atomaren Zahlenwert (Fitnesswert) zuordnet“ (Lindlar, 2012, S.7). Dies stellt eine Herausforderung dar, da gegebenenfalls konkurrierende Zielgrößen in einer Zielfunktion integriert werden müssen. Die verbleibende Fehlerwahrscheinlichkeit kann bei den Optimierungsalgorithmen nicht bestimmt werden, da nicht nachgewiesen werden kann, dass die Optimierung alle Minima beziehungsweise Maxima während der Optimierung findet.

### Statistische Auswertungen

Eine Informationsquelle für die Analyse der Eigenschaften von stationären und beweglichen Elementen der Szene ist die statistische Auswertung. Für die Analyse von stationären Elementen, wie Straßen, können Offline- beziehungsweise Online-Quellen verwendet werden. Lippold

u. a. (2014) oder *OpenStreetMaps*<sup>5</sup> stellen solche Informationsquellen dar. Für die Analyse von beweglichen Elementen können beispielsweise Messfahrten verwendet werden. Mit Hilfe der Informationsquellen können minimale und maximale Werte für die Wertebereiche von verschiedenen Eigenschaften bestimmt werden. Weiterhin können Diskretisierungsstufen durch die Verteilung der identifizierten Werte abgeleitet werden.

Eigenschaften, die beispielsweise für Straßen abgeleitet werden können, sind Kurvenkrümmungen und die Anzahl der Fahrstreifen auf den analysierten Streckenabschnitten. Die Werte können jedoch nur geschätzt werden, da die Qualität der Informationen unmittelbar von der Qualität der zur Verfügung gestellten Quelldaten abhängt. Bei *OpenStreetMaps* werden Karten beispielsweise von Nutzern anhand von verfügbaren Luftbildern gezeichnet. Somit ist die Qualität der Daten abhängig von der Aktualität der Luftbilder und der Genauigkeit der nachgezeichneten Strecken.

Die statistische Auswertung von Straßen aus Offline- und Online-Quellen ist eine Erweiterung der Auswahl von Einflussparametern aus den Normen und Richtlinien, da die Repräsentativität der Informationsquelle überprüft werden kann. Auf diese Weise können die angegebenen minimalen und maximalen Werte der Wertebereiche überprüft werden. Des Weiteren können, wie von Prialé Olivares u. a. (2016) gezeigt, durch die Häufigkeitsverteilung von Werten Diskretisierungsstufen abgeleitet werden. Zusätzlich können durch die Identifikation von Ausreißern beziehungsweise Werten mit einer geringen Häufigkeit pathologische Fälle geschätzt werden. Dadurch wird die Möglichkeit geschaffen, Fälle für die Validierung zu finden, die gegebenenfalls von den Funktionsentwicklern nicht beachtet wurden, da diese nur mit einer geringen Wahrscheinlichkeit auftreten. Bei Werten mit einer hohen Wahrscheinlichkeit ist davon auszugehen, dass diese auch den Funktionsentwicklern bekannt sind. Bei der Verwendung von statistischen Auswertungen ist wiederum zu beachten, dass die Herausforderung der repräsentativen Stichprobe entsteht, die bereits in Kapitel 7.1.2 diskutiert wurden. Über die verbleibende Fehlerwahrscheinlichkeit der Diskretisierung der wertkontinuierlichen Einflussparameter wird nur bedingt eine Aussage getroffen, da nur Diskretisierungsstufen getestet werden, die in der Realität existieren und erfasst wurden. Damit wird keine Aussage über Straßen getroffen, die in Zukunft gebaut werden oder nicht erfasst wurden.

## Expertenwissen

Expertenwissen kann ebenfalls für die Analyse der Eigenschaften der Elemente eingesetzt werden. Durch die langjährigen Erfahrungen mit dem Testobjekt können Experten abschätzen, welche Diskretisierungsstufen bei einem Test zu einem Fehler des Systems führen könnten. Jedoch sind die Entscheidungen für die Wahl der Parameterstufen meist intuitiv und nicht systematisch ableitbar (Eltaher, 2013, S.4). Somit sollten Methoden entwickelt werden, um das Wissen der Experten zu objektivieren und für einen systematischen Ansatz, wie den modularen virtuellen Testbaukasten, nutzbar zu machen. Auch für die Analyse der Eigenschaften ist zu beachten, dass das Expertenwissen nur bei einer evolutionären Entwicklung verfügbar ist. Bei einer disruptiven Entwicklung steht kein Expertenwissen zur Verfügung.

---

<sup>5</sup><http://www.openstreetmap.org/>, abgerufen: 17.03.2016

## 7.5 Schema zur Darstellung von Einflussparametern

Dieses Unterkapitel stellt ein Schema zur Darstellung von Einflussparametern vor, welches im modularen virtuellen Testbaukasten verwendet wird. Zunächst wird das Schema beschrieben. Daraufgehend wird die Auswahl und Analyse von Einflussparametern sowie das vorgestellte Schema an der in Kapitel 3.3 vorgestellten Fahrfunktion exemplarisch angewendet. Das Unterkapitel schließt mit einer Bewertung der gestellten Anforderungen ab.

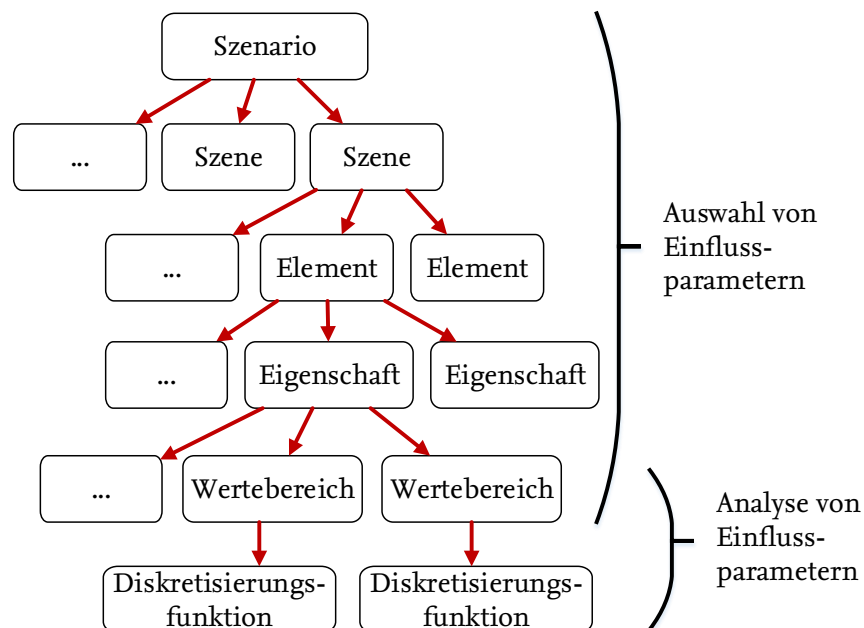
### 7.5.1 Beschreibung des Schemas

Das Schema orientiert sich an den von Grochtmann und Grimm (1993) vorgestellten Klassifikationsbäumen und den von Thomason und Gonzalez (1985, S.26f) vorgestellten Szenenbäumen. Die Baumstruktur wird in beiden Anwendungen für eine hierarchische Darstellung verwendet. Thomason und Gonzalez (1985, S.26f) beschreiben in den Szenenbäumen die räumliche Anordnung von Objekten. Grochtmann und Grimm (1993) nutzen die Baumstrukturen für die Darstellung von diskreten Testparametern.

Die Anwendungen zeigen, dass eine Baumstruktur den Vorteil bietet, dass sich die Parameter strukturiert und hierarchisch beschreiben lassen. Eine solche Darstellungsmöglichkeit wird ebenfalls für das Schema zur Darstellung von Einflussparametern benötigt, um verschiedene Abstraktionsebenen und damit Detaillierungsgrade zu betrachten. Die vorgestellten Klassifikationsbäume von Grochtmann und Grimm (1993) bieten das Potential, die Anforderungen an das Schema zu erfüllen. Jedoch erfordern die Klassifikationsbäume für die Anwendung diskretisierte Parameterstufen am Ende eines Astes. Eine Darstellung von Diskretisierungsmöglichkeiten für die verschiedenen Wertebereiche ist im Ansatz nicht vorgesehen. Daher wird im Folgenden der Ansatz von Grochtmann und Grimm (1993) für das Schema zur Beschreibung der Einflussparameter im modularen virtuellen Testbaukasten angepasst. Dazu werden die Stufen des Baumes an die Terminologie aus Kapitel 2.3 angeglichen und um Wertebereiche und Diskretisierungsfunktionen erweitert.

Für eine systematische Beschreibung der Einflussparameter wird das Schema verwendet, welches in Abbildung 7.1 dargestellt ist. In der Abbildung ist neben dem Schema zu sehen, welche Ebenen des Schemas von der Auswahl der Einflussparameter und welche Ebenen von der Analyse der Einflussparameter befüllt werden. Das Schema wird durch einen hierarchischen Aufbau beschrieben. Auf der obersten Hierarchieebene steht das Szenario, welches nach Kapitel 2.3 durch eine Abfolge von Szenen definiert ist. Damit ergibt sich eine 1-zu- $n$ -Beziehung zwischen dem Szenario und den beinhalteten Szenen. Jede Szene beinhaltet  $m$  Elemente. Diese Elemente können beweglich, wie Verkehrsteilnehmer, oder stationär, wie Fahrstreifen, sein. Jedes Element hat  $o$  Eigenschaften zur Beschreibung. Bei diesen Eigenschaften handelt es sich um die Einflussparameter, die von der Auswahl identifiziert wurden. Als Beispiel für einen Fahrstreifen könnte die Fahrstreifenbreite oder der Markierungstyp genannt werden. Jede Eigenschaft hat  $k$  Wertebereiche, die sich nicht widersprechen dürfen. Der Wertebereich kann diskret sein, um beispielsweise eine Objektklasse zu beschreiben, oder durch einen kontinuierlichen Wertebereich mit einer oberen und unteren Grenze beschrieben werden. Jeder kontinuierliche Wertebereich wird durch eine Diskretisierungsfunktion beschrieben. Die Diskretisierungsfunktion ist eine funktionale Beschreibung der Größe der Diskretisierungsstufen über den jeweiligen Wertebereich. Diese kann beispielsweise eine lineare Diskretisierungsfunktion mit einer äquidistanten Schrittweite sein, die von der Anzahl der Diskretisierungsstufen und den Grenzen des Wertebereichs abhängig ist. Der Vorteil





**Abbildung 7.1:** Schema zur Beschreibung von Einflussparametern

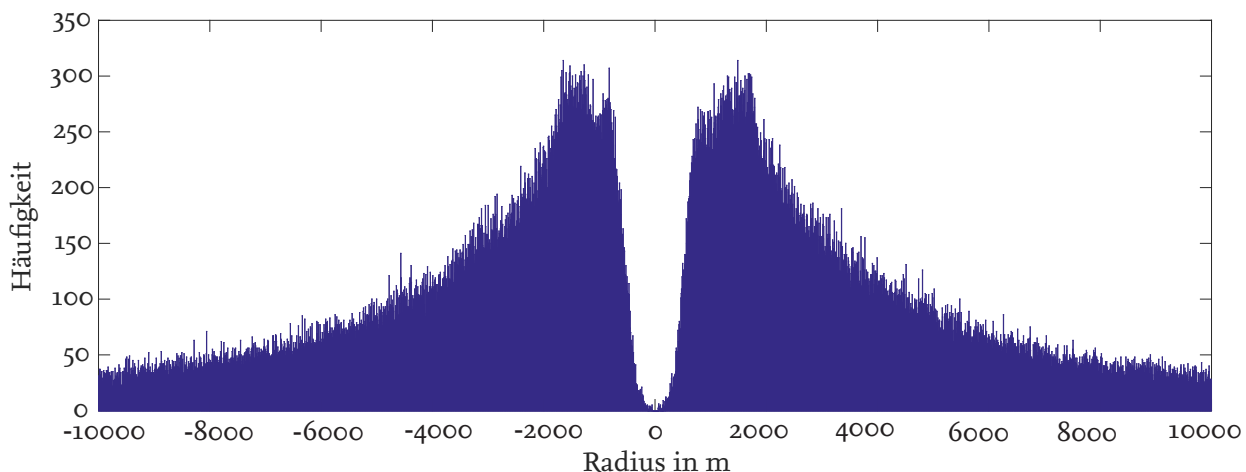
dieser Diskretisierung besteht darin, dass für die jeweiligen Wertebereiche veränderbare Diskretisierungsstufen und damit unterschiedliche Auflösungen erstellbar sind. Dadurch ist es möglich, statistische Verteilungen, die durch die Analyse der Einflussparameter bestimmt wurden, zu beschreiben.

Durch die Auswahl der Einflussparameter werden die Hierarchieebenen Szenario, Szene, Element, Eigenschaft und Wertebereich beschrieben. Mit Hilfe der Analyse der Einflussparameter werden für die Wertebereiche die Diskretisierungsfunktionen, beispielsweise durch statistische Verteilungen, für das Schema erstellt.

### 7.5.2 Exemplarische Anwendung des Schemas an einer aktiven Querführungsfunktion

Im Folgenden wird die Auswahl und Analyse von Einflussparametern und das Schema zur Darstellung der Parameter anhand der in Kapitel 3.3 vorgestellten Fahrfunktion exemplarisch angewendet. Das vorgestellte System übernimmt die Querführung des Fahrzeugs auf Autobahnen und führt das Fahrzeug im eigenen Fahrstreifen. Als exemplarische Informationsquelle werden die Normen und Richtlinien für die Anlage von Autobahnen (FGSV, 2008) verwendet. Aus diesen lassen sich die Parameter zur Beschreibung der Fahrstreifen auf Autobahnen herleiten. Als Parameter werden die Fahrstreifenbreite, die Krümmung der Straße, der Typ der Fahrstreifenmarkierung und die Breite der Fahrstreifenmarkierung auf der linken und rechten Seite ausgewählt. Für die Auswahl der Wertebereiche der Fahrstreifenbreiten werden die Regelquerschnitte verwendet, die in Kapitel 2.4 vorgestellt wurden. Aus diesen lässt sich ableiten, dass die Fahrstreifenbreite auf Autobahnen zwischen 3,25 Meter und 3,75 Meter liegt. Als Diskretisierungsfunktion wird eine lineare Funktion mit einer Diskretisierungsstufe von 0,25 Meter gewählt. Damit werden die Fahrstreifenbreiten getestet, die in den Regelquerschnitten vorhanden sind. Mögliche Diskretisierungsstufen für den Typ der Fahrstreifenmarkierung werden ebenfalls in den Normen und Richtlinien angegeben.

Für die Fahrstreifenmarkierung werden in den Normen und Richtlinien (FGSV, 1980) die Typen durchgezogen und unterbrochen genannt. Für die unterbrochene Markierung wird eine Strichlänge von 6 Meter angegeben. Die Lücken werden mit den Werten 12 Meter und 3 Meter beschrieben. Die Breite der Markierung wird mit 0,15 Meter und 0,30 Meter angegeben. Für die Parameter Markierungstyp und Markierungsbreite sind aufgrund der diskreten Werte keine weiteren Diskretisierungen notwendig. Für die Radien der Kurven werden je nach Autobahntyp minimale Werte zwischen 280 und 900 Meter angegeben. Diskretisierungsstufen können aus den Normen und Richtlinien nicht abgeleitet werden. Daher ist ein weiterer Schritt durch die Analyse notwendig. Hierfür kann beispielsweise die statistische Auswertung verwendet werden. Abbildung 7.2 zeigt ein Histogramm einer Schätzung der statistischen Verteilung von auftretenden Kurvenradien der deutschen Autobahnen A1 - A9<sup>6</sup>. Die Radien wurden aus der Online-Quelle *OpenStreetMaps* extrahiert. Kurvenradien, die größer als 10 Kilometer sind, wurden als Geraden interpretiert und nicht in die Abbildung aufgenommen. Das Histogramm besitzt eine Auflösung von 5 Meter pro Säule.



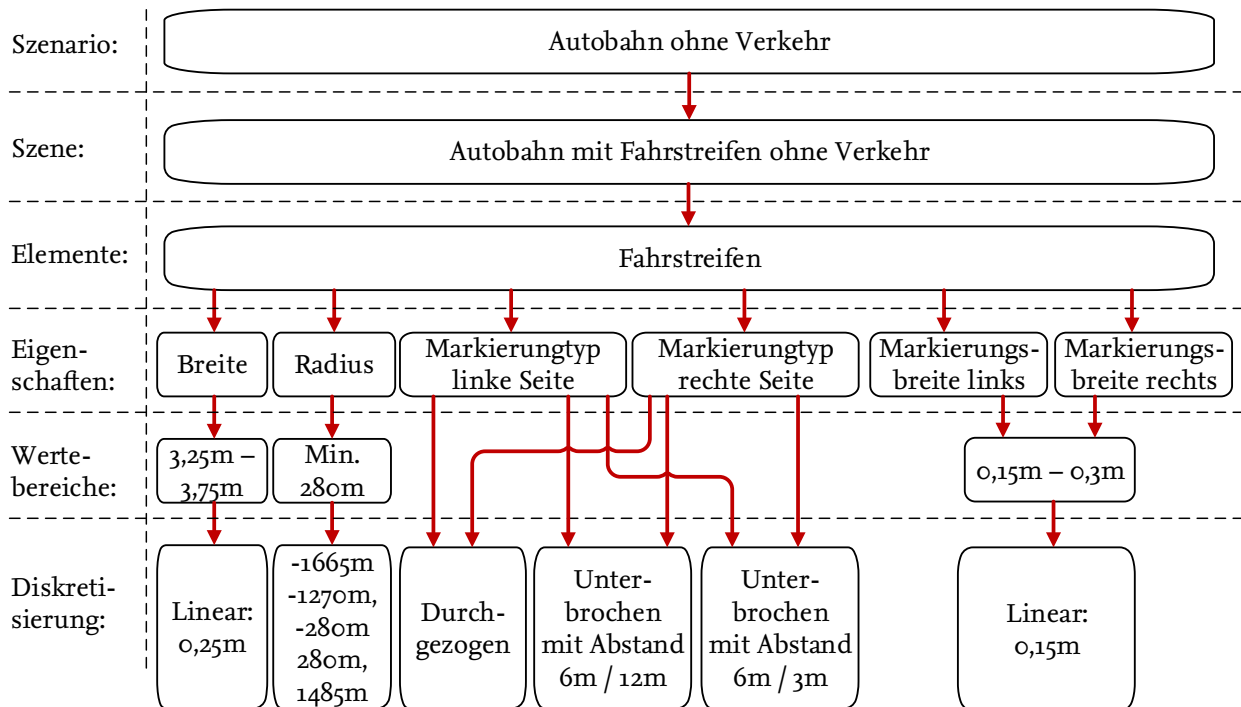
**Abbildung 7.2:** Statistische Verteilung der Radien der Autobahnen A1 - A9

Mit der statistischen Verteilung besteht die Möglichkeit, beispielsweise mit dem Ansatz von Prialé Olivares u. a. (2016), eine Verteilungsfunktion zu bestimmen und daraus abgetastete Diskretisierungsstufen abzuleiten. Dieser Schritt wird für die exemplarische Anwendung der Auswahl und Analyse nicht durchgeführt, da beispielsweise die Wahl der Abtaststrategie für die Verteilung noch eine offene Forschungsfrage darstellt und vom Testbaukasten nicht beantwortet wird. Für die weitere Betrachtung der Anwendung des Testbaukastens werden exemplarisch die minimal erlaubten Radien von 280 Meter und -280 Meter für Links- und Rechtskurven sowie die drei Radien ausgewählt, die am häufigsten auftreten. Dies sind

<sup>6</sup>Für die Erhebung der Daten wurden die Verläufe der deutschen Autobahnen aus OpenStreetMaps extrahiert. Auf dieser Basis wurde für jede Autobahn ein kubischer Spline berechnet. Die Splines wurden mit einem Abstand von 10 Meter abgetastet, um die Krümmung des Splines in den jeweiligen Punkten zu berechnen. Ein Fitting in Klothoiden wurde nicht durchgeführt. Diese Analyse ist in weiteren Forschungsarbeiten zu untersuchen. Zur Überprüfung der Plausibilität der Daten wurde die Gesamtlänge der Autobahnen verglichen. Laut Wikipedia ([https://de.wikipedia.org/wiki/Liste\\_der\\_Bundesautobahnen\\_in\\_Deutschland](https://de.wikipedia.org/wiki/Liste_der_Bundesautobahnen_in_Deutschland), abgerufen 25.06.2016) ergibt sich eine Gesamtlänge von 5495 Kilometer für die Autobahnen A1 - A9. Die Gesamtlänge der Splines beträgt 5486 Kilometer. Damit ergibt sich ein Fehler von 0,16%.



die Werte -1665 Meter, -1270 Meter und 1485 Meter. Abbildung 7.3 zeigt das ausgefüllte Schema für die Auswahl und Analyse der exemplarischen Fahrfunktion.



**Abbildung 7.3:** Ausgefülltes Schema für die exemplarische Querführungsfunktion

### 7.5.3 Bewertung der gestellten Anforderungen

Durch den Aufbau des Schemas und die Erweiterung der Klassifikationsbäume von Grochtmann und Grimm (1993) wird die Anforderung an die Beschreibung von Wertebereichen und Diskretisierungsfunktionen erfüllt. Für die Wertebereiche und Diskretisierungsfunktionen wurde jeweils eine Ebene im Schema vorgesehen, wodurch für jede Eigenschaft eines Elements sowohl verschiedene Wertebereiche als auch Diskretisierungsfunktionen beschreibbar sind. Das exemplarisch ausgefüllte Schema für die Querführungsfunktion, welches in Abbildung 7.3 dargestellt ist, zeigt die Möglichkeit, Wertebereiche und Diskretisierungsfunktionen zu definieren. Die Anforderung nach der Betrachtung verschiedenster Arten von Informationsquellen wird, bezogen auf die vorgestellten Informationsquellen, erfüllt. Da die vorgestellten Informationsquellen entweder Elemente der Szene, Eigenschaften eines Elements, Wertebereiche oder Diskretisierungsstufen bereitstellen, können die Daten der Informationsquellen mit dem vorgestellten Schema erfasst werden. Jedoch kann keine allgemeine Aussage über die Erfüllung der Anforderungen getroffen werden, da die vorgestellten Informationsquellen nicht alle möglichen Informationsquellen darstellen. Daher wird die Anforderung nur bezogen auf die vorgestellten Informationsquellen als erfüllt angesehen. Durch die Baumstruktur des Schemas kann die Darstellung der Einflussparameter formalisiert werden. Weiterhin kann das Schema beispielsweise in Form einer XML-Struktur<sup>7</sup> gespeichert und an das Modul der

<sup>7</sup>XML: engl. extensible markup language (Frei übersetzt: erweiterbare Auszeichnungssprache) beschreibt die Darstellung hierarchisch strukturierter Daten in Form von Textdateien.

Testfallerstellung übergeben werden. Damit werden die gestellten Anforderungen unter der oben genannten Bedingung vom vorgestellten Schema erfüllt.

## 7.6 Zusammenfassung

Im vergangenen Kapitel wurde das Modul der Auswahl und Analyse von Einflussparametern beschrieben. Dazu wurden zunächst Anforderungen an die Auswahl und Analyse von Einflussparametern definiert. Daraufhin wurde ein ausgewählter Teil aus dem Stand der Technik zur Auswahl und Analyse von Einflussparametern in Testkonzepten vorgestellt und anhand der gestellten Anforderungen bewertet. Die Auswahl der Literatur zeigt, dass die Auswahl und Analyse von Einflussparametern derzeit noch wenig Beachtung findet. Viele Testkonzepte erwarten bereits diskretisierte Werte für die Testerstellung. Welchen Ursprung die gewählten Parameter und diskretisierten Werte haben, wird nicht beschrieben. Verschiedene Informationsquellen zur Auswahl und Analyse wurden in den bisherigen Testkonzepten nicht beachtet.

Daher wurden in diesem Kapitel mögliche Informationsquellen zur Auswahl und Analyse von Einflussparametern vorgestellt und anhand der gestellten Anforderungen bewertet. Bei der Betrachtung der verschiedenen Informationsquellen zeigen sich individuelle Stärken und Schwächen. Die Tabellen 7.1, 7.2 und 7.3 fassen die Ergebnisse der Bewertung zusammen. Insgesamt zeigt sich, dass die Quellen zahlreiche Beiträge zur Auswahl und Analyse liefern. Jedoch wird auch deutlich, dass die Quellen für die Erfüllung zahlreicher Anforderungen derzeit keinen Beitrag leisten. Dies gilt besonders bei der Analyse von Einflussparametern. So kann die Frage, welche Testabdeckung mit einem Testset erreicht wurde, derzeit nicht beantwortet werden, da keine der analysierten Informationsquellen eine Möglichkeit zur Bestimmung der verbleibenden Fehlerwahrscheinlichkeit bietet. Weiterhin liegt der Fokus bei vielen Informationsquellen vor allem auf dem aktuellen Stand der Einflussparameter. Es wird nicht analysiert, wie sich diese in Zukunft verändern könnten. So ist eine offene Frage: Welches Risiko wird eingegangen, wenn Parameter nicht beachtet und getestet werden. Gleiches gilt für die Normen und Richtlinien. Wie ist mit Gebieten umzugehen, für die keine Normen und Richtlinien existieren? Dies sind offene Fragestellungen, die auch in dieser Arbeit nicht beantwortet werden. Der Testbaukasten leistet bei der Auswahl und Analyse derzeit einen Beitrag durch die Vorstellung eines Schemas zur Darstellung von Parametern und eine Bewertung verschiedener Informationsquellen bezüglich der gestellten Anforderungen. Die Anwendung des Schemas wurde exemplarisch anhand der in Kapitel 3.3 vorgestellten Fahrfunktion gezeigt. Einen Ansatz für die Diskretisierung und die Abschätzung der Fehlerwahrscheinlichkeit liefert die vorliegende Arbeit und der modulare virtuelle Testbaukasten derzeit nicht. Jedoch sollen die Herausforderungen der Diskretisierung und die daraus resultierende Fehlerwahrscheinlichkeit an dieser Stelle thematisiert werden, um weitere Forschungsarbeiten auf diesem Gebiet zu fördern.

Anforderung: Quelle:	Identifikation von Einflussparametern	Identifikation von repräsentativen Parameterbereichen
Funktions- spezifikation & Anwendungs- szenarien	+ Einflussparameter beschrieben – Fehlerhafte, unvollständige Spezifikationen	+ Einflussparameterbereiche beschrieben – Fehlerhafte, unvollständige Spezifikationen
4-Ebenen- Modell	+ Iterative Erweiterung möglich – Immer unvollständig gegenüber der Realität	+ Iterative Erweiterung möglich
Normen & Richtlinien	+ Parameter für stationäre Elemente – Nur in Ländern mit Normen & Richtlinien verfügbar	+ Maximal- und Minimalwerte zum Teil verfügbar – Vergleich Normen und Realität
Fahrmanö- vertheorien	+ Parameter für bewegliche Elemente	– Regionale Abhängigkeiten der Parameter
Unfallanalysen	+ Parameter für stationäre und bewegliche Elemente – Nur die Erfassung von Unfällen – Übertragbarkeit auf automatisierte Fahrfunktionen unklar	+ Exakte Parameterbereiche für stationäre und bewegliche Elemente – Nur die Erfassung von Unfällen
Expertenwissen bei evolutionärer Entwicklung	+ Parameter für stationäre und bewegliche Elemente – Wissen nur implizit vorhanden	+ Parameterbereiche für stationäre und bewegliche Elemente – Wissen nur implizit vorhanden

**Tabelle 7.1:** Auswertung der Erfüllung der Anforderungen durch die Informationsquellen für die Auswahl von Einflussparametern (Fortsetzung nächste Seite)

Anforderung: Quelle:	Identifikation von Einflussparametern	Identifikation von repräsentativen Parameterbereichen
Messfahrten im öffentlichen Straßenverkehr	<ul style="list-style-type: none"> <li>+ Identifikation durch Auswertung der Fahrten möglich</li> <li>– Aufwendige Auswertung notwendig</li> <li>– Ergebnisse abhängig von der Stichprobe</li> </ul>	<ul style="list-style-type: none"> <li>+ Identifikation durch Auswertung der Fahrten möglich</li> <li>– Aufwendige Auswertung notwendig</li> <li>– Ergebnisse abhängig von der Stichprobe</li> </ul>
Fahrzeugkataloge	<ul style="list-style-type: none"> <li>+ Parameter für bewegliche Fahrzeuge</li> </ul>	<ul style="list-style-type: none"> <li>+ Theoretisch mögliche Werte abschätzbar</li> </ul>

**Tabelle 7.2:** Auswertung der Erfüllung der Anforderungen durch die Informationsquellen für die Auswahl von Einflussparametern (Fortsetzung)

Anforderung: Quelle:	Bestimmung von Diskretisierungsstufen	Bestimmung der Fehlerwahrscheinlichkeit durch die Diskretisierung
Optimierungsalgorithmen	<ul style="list-style-type: none"> <li>+ Selbständiges Suchen von Diskretisierungsstufen</li> <li>– Nur anwendbar, wenn viele Testfälle durchgeführt werden können</li> </ul>	<ul style="list-style-type: none"> <li>– Keine Aussagen möglich</li> </ul>
Statistische Auswertung von Wertebereichen	<ul style="list-style-type: none"> <li>+ Über Verteilungshistogramm möglich</li> <li>– Für Straßen: Nur Straßen getestet, die vorhanden sind</li> </ul>	<ul style="list-style-type: none"> <li>– Keine Aussagen möglich</li> </ul>
Expertenwissen bei evolutionärer Entwicklung	<ul style="list-style-type: none"> <li>+ Schätzung über Wissen möglich</li> <li>– Nur implizites Wissen</li> </ul>	<ul style="list-style-type: none"> <li>– Keine Aussagen möglich</li> </ul>

**Tabelle 7.3:** Auswertung der Erfüllung der Anforderungen durch die Informationsquellen für die Analyse von Einflussparametern

## 8 Testfallerstellung

Dieses Kapitel beschreibt die Testfallerstellung des modularen virtuellen Testbaukastens. Zunächst werden Anforderungen an die Testfallerstellung definiert. Daraufhin wird der relevante Stand der Technik zur Testfallerstellung beschrieben und auf die Erfüllung der gestellten Anforderungen analysiert. Basierend auf der Analyse der erfüllten Anforderungen wird der Forschungsbedarf abgeleitet. Darauf aufbauend wird ein generisches 4-Ebenen-Modell zur Beschreibung von Szenarien und eine systematische Testfallableitung vorgestellt, die aus logischen Szenarien konkrete Testfälle erstellt.

### 8.1 Anforderungen

Das folgende Unterkapitel beschreibt die Anforderungen an die Testfallerstellung. Diese werden in die Anforderungen an die Beschreibung von Szenarien und an die Testfallableitung für automatisierte Fahrfunktionen unterteilt. Die Anforderungen wurden auf Basis der Erfahrungen des Autors im Projekt des Engstellenassistenten erhoben.

#### 8.1.1 Anforderungen an die Beschreibung von Szenarien für automatisierte Fahrfunktionen

**Die Testfallerstellung muss eine einheitliche Struktur für Szenarien und Testfälle nutzen.**

Die Szenarien und erzeugten Testfälle müssen eine generische Struktur haben, damit die Möglichkeit besteht, Szenarien und Testfälle in verschiedenen Umgebungen, wie Autobahnen, Landstraßen oder Stadtstraßen, zu beschreiben und mit verschiedenen Testwerkzeugen durchzuführen. Idealerweise kann die Struktur bei einer Standardisierung von Szenarien unterstützen, um die Szenarien mit einer Vielzahl von Testwerkzeugen, wie der Simulation, durchzuführen. Die Struktur muss eine Anpassung bestehender Testfälle an neue oder veränderte Testobjekte ermöglichen. Dabei besteht die Forderung, dass die Einflussparameter, die im Schema aus Kapitel 7.5 beschrieben wurden, in der generischen Struktur darstellbar sind. Dafür muss die Struktur auf einer submikroskopischen Ebene beschreibbar sein. Des Weiteren besteht die Anforderung, dass die Struktur erweiterbar ist, damit neue Einflussparameter hinzugefügt werden können.

**Die Testfallerstellung muss Szenarien erstellen, die reproduzierbar durchführbar sind.**

Die beschriebenen Szenarien müssen reproduzierbar durchführbar sein, damit die Ergebnisse von verschiedenen Testdurchläufen vergleichbar sind. Dabei muss sowohl eine globale als auch eine lokale Reproduzierbarkeit, wie sie in Kapitel 5.1.2 durch die Anforderungen an die Effizienz des Testkonzepts gefordert wurde, ermöglicht werden. Dies gilt besonders, wenn sich nicht deterministisch verhaltende Elemente, wie reale Fahrer oder reale Fahrzeuge, im Szenario vorhanden sind. Die Elemente des Szenarios müssen für die lokale Reproduzierbarkeit ihr Verhalten an das Verhalten des Testobjekts anpassen, um die Testfälle aus der Sicht des Testobjekts reproduzierbar durchzuführen.

### 8.1.2 Anforderungen an die Testfallableitung für automatisierte Fahrfunktionen

**Die Testfallerstellung soll die minimale Anzahl von Testfällen ableiten, die für die Abnahme der Fahrfunktion erforderlich ist.**

Von der Testfallerstellung wird gefordert, dass die minimale Anzahl von Testfällen ableitet wird, die erforderlich ist, um das Testobjekt für ausgewählte Szenarien hinreichend für eine Abnahme beziehungsweise Freigabe zu testen.

Diese Anforderung wird in der vorliegenden Arbeit nicht erfüllt. Trotzdem wird die Anforderung genannt, um zu zeigen, dass die Anforderung bei der Erstellung des Konzepts bereits bedacht wurde. Die Definition der minimal erforderlichen Anzahl von Testfällen, die für die Freigabe benötigt wird, ist derzeit noch Gegenstand der Forschung, wie das Forschungsprojekt *PEGASUS*<sup>1</sup> zeigt (Plättner u. a., 2016).

Um einen Forschungsbeitrag zur Erfüllung dieser Anforderung zu leisten, wird in der vorliegenden Arbeit die paarweise Testabdeckung, welche in Kapitel 8.4.3.1 vorgestellt wird, anhand des Engstellenassistenten in Kapitel 12 evaluiert. Damit wird analysiert, welche Effekte der Einflussparameter auf Basis dieser minimierten Anzahl von Testfällen abgeleitet werden können.

**Die von der Testfallerstellung abgeleiteten Testfälle sollen minimal redundant sein.**

Die abgeleiteten Testfälle sollen eine minimale Redundanz haben. In der vorliegenden Arbeit bedeutet dies, dass die Testfälle mit den selben Kombinationen der Diskretisierungsstufen der Einflussparameter minimiert und getestet werden. Dabei darf die Anzahl der Parameter und der Diskretisierungsstufen, die für die Testfallableitung beachtet werden, nicht begrenzt sein. Die Durchführung von redundanten Testfällen führt zu keiner weiteren Erhöhung der Testabdeckung. Aus diesem Grund ist es für die Testfallableitung besonders wichtig, die Anzahl von redundanten Testfällen zu minimieren. Mit der Erfüllung dieser Anforderung wird ein Beitrag zur Anforderung der Ökonomie an das Testkonzept geleistet.

**Die Testfallerstellung muss die Testfälle ohne Informationen über den Quellcode ableiten.**

Die Ableitung von Testfällen muss ohne Informationen über den Quellcode möglich sein. Zur Testfallableitung dürfen lediglich die Szenariobeschreibung und die identifizierten Parameterstufen verwendet werden. Daher darf der Testbaukasten ausschließlich Methoden des Black-Box-Testens verwendet, wie sie in Kapitel 4.1.1 vorgestellt wurden.

## 8.2 Stand der Technik

Dieses Kapitel beschreibt den relevanten Stand der Technik zur Testfallerstellung. Dafür werden existierende Modelle zur Beschreibung von Szenarien präsentiert. Der ausgewählte Stand der Technik bezieht sich auf Ansätze, die Szenarien für die Verkehrssimulation beschreiben und bei denen Fahrfunktionen getestet werden. Weiterhin werden Ansätze zur Testfallableitung aus logischen Szenarien für automatisierte Fahrfunktionen vorgestellt und anhand der Anforderungen bewertet. Dabei bezieht sich der ausgewählte Stand der Technik

<sup>1</sup>Projekt zur Etablierung von generell akzeptierten Gütekriterien, Werkzeugen und Methoden sowie Szenarien und Situationen zur Freigabe hochautomatisierter Fahrfunktionen

auf Arbeiten, die Testfälle für automatisierte Fahrfunktionen ableiten und anschließend in der Simulation durchführen.

### 8.2.1 Beschreibung von Szenarien

Schmidt (2012) präsentiert einen Ansatz für die Beschreibung von Szenarien für kamera-basierte Fahrerassistenzsysteme. Der Fokus liegt auf der Erstellung von fotorealistischen Bildern. Schmidt (2012, S.188f) beschreibt die Szenarien in einem *TESTE*-Schema mit fünf Kategorien: 1. Strecke als Straßenbeschreibung, 2. Umwelt in Form von Wetter und der Tageszeit, 3. Umgebung als die Randbebauung der Straße, 4. Objekte um das Ego-Fahrzeug und 5. Beschreibung des Ego-Fahrzeugs. Die Anfangsbuchstaben der englischen Übersetzungen des ersten Worts jeder Kategorie ergeben das Wort *TESTE*<sup>2</sup>.

Das *TESTE*-Schema besitzt das Potential, eine einheitliche Struktur für Szenarien und Testfälle darzustellen, da Schmidt (2012, S.152f) verschiedene Ebenen für die Beschreibung des Szenarios einführt und für die Beschreibung der Straße das standardisierte *OpenDrive*-Format (Dupuis u. a., 2010b) verwendet. Dadurch besteht die Möglichkeit, submikroskopische Szenarien für verschiedene Umgebungen zu beschreiben. Jedoch nimmt das Ego-Fahrzeug durch die eigene Kategorie im Schema eine besondere Rolle ein. Dies bedeutet, dass das Testobjekt in nur einem Fahrzeug darstellbar ist. Ein Szenario für kooperative Funktionen, bei deren Tests das Testobjekt in mehreren Fahrzeugen gleichzeitig vorhanden ist, und der Test einer automatisierten Lichtsignalanlage zur Koordination des Verkehrsflusses kann mit dem *TESTE*-Schema nicht beschrieben werden. Daher sollte das Ego-Fahrzeug auf keiner Ebene eines Modells explizit spezifiziert werden, sondern als ein unbewegliches beziehungsweise bewegliches Element im Szenario definiert werden, bei dem die automatisierte Fahrfunktion oder das Testobjekt integriert wird. Damit kann das Testobjekt auf verschiedene Elemente im Szenario angewendet werden. Schmidt (2012, S.178f) fordert ebenfalls eine Reproduzierbarkeit der Szenarien. Jedoch bezieht dieser die Reproduzierbarkeit lediglich auf die Fahrdynamikmodelle und die dargestellten Bilder für die Kamera. Ein reproduzierbares Verhalten der unbeweglichen und beweglichen Elemente im Szenario aus Sicht des Testobjekts wird nicht thematisiert, sodass die Anforderung nach der Reproduzierbarkeit mit dem vorgestellten Ansatz nicht erfüllt wird.

Shimada u. a. (2015) nutzen das von Yuen u. a. (2008) vorgestellte hierarchische Modell *Local Dynamic Map* als Umfeldmodell für ein kooperatives Sicherheitssystem, welches die Kommunikationsarten Fahrzeug-zu-Fahrzeug und Fahrzeug-zu-Infrastruktur für eine Kollisionserkennung nutzt. Dabei wird untersucht, ob die Antwortzeiten des Modells den Anforderungen des Sicherheitssystems genügen. Das Umfeldmodell besteht aus vier Ebenen. Auf der ersten Ebene befindet sich die Karte inklusive aller stationären Elemente, d.h Straßen und Randbebauung der Straße. Auf der zweiten Ebene werden relativ zur ersten Ebene alle stationären Elemente, die nicht in der ersten Ebene vorhanden sind, dargestellt. Dies stellt die Verkehrsinfrastruktur dar, die beispielsweise durch Verkehrszeichen, Lichtsignalanlagen oder Landmarken dargestellt wird. Auf der dritten Ebene werden Informationen über alle transient dynamischen Elemente, wie das Wetter, hinterlegt. Die vierte Ebene beschreibt alle dauerhaft dynamischen Elemente, wie die übrigen Verkehrsteilnehmer. Die Ebenen sind nach der Aktualität der Informationen aufgeteilt. Auf den unteren Ebenen haben die Daten

---

<sup>2</sup>TESTE: 1.track, 2.environment, 3.surrounding, 4.test-relevant objects, 5.ego-vehicle.



ein lange Aktualität, da sich die Daten nur selten ändern, während die Daten auf den oberen Ebenen nur in einer kurzen Zeitspanne aktuell sind.

Das vorgestellte Modell besitzt einen strukturierten Aufbau. Jedoch ist die Strukturierung des Modells nach der Aktualität der Daten nur bedingt für die Beschreibung von logischen Szenarien für die Erstellung von konkreten Testfällen aus globaler Sicht verwendbar. Aus globaler Sicht sind die Informationen über die Elemente dauerhaft aktuell, da diese vom Anwender für die jeweiligen Szenen definiert werden. Weiterhin bietet das Modell lediglich eine Beschreibung der Szenarien auf mesoskopischer Ebene. So wird beispielsweise für einen Fahrstreifen lediglich eine Breite angegeben. In der Anwendung von Shimada u. a. (2015) wird das Modell für die Aggregation von Daten aus der Sicht der Testobjekte verwendet. Das Modell wird zur Laufzeit mit den wahrgenommenen Informationen der Testobjekte parametrisiert. Damit ist die Beschreibung der Ebenen immer auf die eingeschränkte Sicht der Testobjekte bezogen und nicht aus der globalen Sicht. Für die Beschreibung von Szenarien für die Erstellung von Testfällen wird eine Beschreibungsform benötigt, die eine globale Beschreibung der Szenen ermöglicht. Dies ist besonders für das Verhalten der beweglichen Verkehrsteilnehmer relevant, die in einer zeit- und/oder ortsgebundener Weise aus globaler Sicht beschreibbar sein sollten. Das von Yuen u. a. (2008) vorgestellte Modell ermöglicht keine globale Beschreibungsform, wodurch die Anforderung nach einem strukturierten Aufbau nicht komplett erfüllt wird. Die Reproduzierbarkeit aus globaler Sicht oder aus Sicht des Testobjektes wird von Yuen u. a. (2008) nicht weiter betrachtet, sodass die Anforderung an die Reproduzierbarkeit nicht erfüllt wird.

Berger (2010, Kap.4.4) stellt ebenfalls ein Modell zur Beschreibung von Szenarien vor. Das Modell trennt den stationären und dynamischen Teil eines Szenarios. Der stationäre Teil wird durch Straßen und die Straßenumgebung definiert. Die Straßen werden durch Fahrstreifen beschrieben, die wiederum durch Punkte, Klothoiden oder Kreise dargestellt werden. Weiterhin werden an dieser Stelle Verkehrszeichen und Lichtsignalanlagen definiert. Die Straßenumgebung wird durch Polygone, Zylinder oder komplexe Modelle abgebildet. Der dynamische Teil des Szenarios wird durch Objekte beschrieben, welche in Form eines Polygons, Rechtecks oder eines komplexen Modells dargestellt werden. Das Verhalten der Objekte kann durch externe Fahrer bestimmt werden. Dies können sowohl menschliche Fahrer als auch eine automatisierte Fahrfunktion sein. Weiterhin besteht die Möglichkeit, eine Liste von Punkten zu definieren, denen das Objekt mit einer konstanten Geschwindigkeit oder Beschleunigung folgt. Berger (2010) stellt durch die Trennung von stationären und dynamischen Teilen eines Szenarios einen strukturierten Ansatz zur Beschreibung von Szenarien vor. Jedoch wird für die Beschreibung der Szenerie kein standardisiertes Format, wie das *OpenDrive*-Format, verwendet. Zudem findet die Beschreibung der dynamischen Objekte aus globaler Sicht und nicht relativ zum Testobjekt statt. Dadurch kann keine lokale Reproduzierbarkeit erreicht werden, da das Verhalten der Objekte zu Beginn des Szenarios fest definiert ist und somit nicht an das Verhalten des Testobjekts anpassbar ist.

Berger u. a. (2014b) stellen einen Szenariogenerator vor, welcher auf Basis einer Szenariobeschreibung ein ausführbares Szenario für die Simulationssoftware *Virtual Test Drive* erstellt. In der Szenariobeschreibung werden der virtuelle Fahrer, die Fahrdynamik und die Trajektorien der beweglichen Elemente definiert. Die Trajektorien werden aus globaler Sicht oder relativ zur Straße beschrieben. Die Szenerie wird im *OpenDrive*-Format als fester Bestandteil des Szenarios in die Szenariobeschreibung inkludiert. Eine Veränderung der Sze-

nerie ist durch die Darstellung des Szenarios nicht möglich. Damit wird die Anforderung nach einem strukturierten Aufbau nur bedingt erfüllt, da die Einflussparameter der stationären Szenerie nicht mit dem Modell veränderbar sind. Die Reproduzierbarkeit von Szenarien ist aufgrund der Beschreibungsform der beweglichen Elemente aus globaler Sicht möglich. Eine lokale Reproduzierbarkeit wird nicht erreicht, da die Beschreibung der Trajektorien nicht mit dem Verhalten des Testobjekts kombinierbar ist. Daher wird die Anforderung an die Reproduzierbarkeit nicht vollständig erfüllt.

Neben den vorgestellten Ansätzen besitzen die verschiedenen Verkehrssimulationen, wie *Virtual Test Drive* (von Neumann-Cosel, 2013), *PreScan* (Tideman, 2010) oder *CarMaker*<sup>3</sup>, eigene Formate für die Beschreibung der Szenarien. Diese sind jedoch nur bedingt öffentlich verfügbar. Eine standardisierte Beschreibungsform von Szenarien ist derzeit nicht verfügbar. Aktuell ist lediglich das standardisierte *OpenDrive*-Format vorhanden, welches Straßennetze beschreibt. Ein Ansatz für ein standardisiertes Beschreibungsformat für Szenarien ist das *OpenScenario*-Format, welches von Schuldt u. a. (2015a) diskutiert wurde und derzeit im Forschungsprojekt *PEGASUS*<sup>4</sup> (Plättner u. a., 2016) weiterentwickelt wird. Das Format soll die Erfüllung der gestellten Anforderungen ermöglichen.

### 8.2.2 Testfallableitung für automatisierte Fahrfunktionen

Schmidt (2012, S.196) schlägt die Nutzung von Äquivalenzklassen und Grenzwertanalysen vor, um Testfälle aus Szenarien abzuleiten. Damit ist mit den gewählten Ansätzen eine Ableitung von Testfällen ohne Quellcode möglich. Die Herausforderung der Diskretisierung von wertkontinuierlichen Parametern, die für die Erstellung der Äquivalenzklassen benötigt werden, und die aus der Diskretisierung resultierende Fehlerwahrscheinlichkeit, die bereits in Kapitel 7.1 thematisiert wurde, werden nicht diskutiert. Für die Bestimmung der Reihenfolge von Szenarien nutzt Schmidt (2012, S.199) kombinatorische Algorithmen. Die Algorithmen werden verwendet, um auf Szenarioebene ähnliche Szenarien für den Test aneinanderzureihen und so die Veränderung der Parameter von einem Szenario zum nächsten Szenario zu minimieren. Dadurch wird die Anzahl der Übergänge zwischen den verschiedenen Szenarien reduziert und die Anzahl von redundant durchgeführten Szenarien minimiert. Die kombinatorischen Methoden werden jedoch nicht für die Variation der Diskretisierungsstufen der Parameter innerhalb eines Szenarios verwendet.

Hilf u. a. (2010) und Tatar und Mauss (2014) präsentieren einen Ansatz für die Testfallableitung von Szenarien mit diskreten Parameterstufen. Der Ansatz wird anhand einer Seitenwindstabilisierungsfunktion mit den Eingangsparametern: Querneigungswinkel der Straße, Windeigenschaften, Fahrzeugeingangsdaten wie Beschleunigung, Bremsdruck und Lenkwinkel gezeigt. Basierend auf den Eingangsdaten und vorher durchgeführten Testfällen werden neue Testfälle durch den Testfallgenerator erzeugt. Das Ziel des Testfallgenerators ist die Erzeugung von Testfällen, um das System in allen möglichen Systemzuständen mit mindestens einem Testfall zu testen. Dabei sollen Systemzustände aufgedeckt werden, bei denen das System seine Spezifikation verletzt. Die Teststrategie wird von Hilf u. a. (2010) verglichen mit einem Schachspiel. Die Testergebnisse werden mit vordefinierten Konditionen für die Ausgangsparameter verglichen. Die Testabdeckung wird mit Hilfe der erreichten

<sup>3</sup><http://ipg.de/de/simulationsolutions/carmaker/>, abgerufen: 17.03.2016

<sup>4</sup>PEGASUS: Projekt zur Etablierung von generell akzeptierten Gütekriterien, Werkzeugen und Methoden sowie Szenarien und Situationen zur Freigabe hochautomatisierter Fahrfunktionen

Systemzustände gemessen. Der Ansatz ermöglicht eine Testfallableitung ohne Informationen über den Quellcode des Testobjekts. Durch die gewählte Teststrategie wird jedoch nicht gezeigt, dass die Anzahl der redundanten Testfälle vor der Testdurchführung minimiert wird, da die Tests iterativ auf Basis der Testergebnisse aufgebaut werden und nicht zum Start der Testdurchführung feststehen. Damit besteht die Gefahr, dass redundante Testfälle zur Laufzeit durchgeführt werden.

Lindlar (2012) stellt eine Testableitung durch einen modellbasierten evolutionären Funktionstest vor. Dabei werden auf Basis einer Fitnessfunktion, welche die Testergebnisse bewertet, und evolutionären Algorithmen neue Testfälle abgeleitet (Lindlar, 2012, S.5,7,40). Die Tests werden ohne Informationen über den Quellcode erstellt. Bei diesem Ansatz wird jedoch nicht gezeigt, dass die Anzahl der redundanten Testfälle vor der Testdurchführung minimiert wird, da die Testfälle, wie beim Ansatz von Tatar und Mauss (2014), iterativ erstellt werden. Damit kann nicht gezeigt werden, dass die Anzahl der redundanten Testfälle minimiert wird, da solange Testfälle durchgeführt werden bis der evolutionäre Algorithmus in ein Extremum der Fitnessfunktion konvertiert oder die maximale Anzahl von vorgegebenen Testfällen durchgeführt wurden.

Prialé Olivares u. a. (2016) stellen einen Ansatz für die Ableitung von Strecken für Assistenzfunktionen auf Basis von stochastischen Verteilungen für die Simulation vor. Hierfür werden Markov-Ketten und Monte-Carlo Methoden verwendet, um aus statistischen Verteilungen den Verlauf einer virtuellen Strecke zu generieren. In einem Fallbeispiel wird gezeigt, wie aus einer Verteilung der Krümmung von realen Strecken aus *OpenStreetMaps* eine virtuelle Strecke erstellt wird. Weitere Bestandteile werden der Strecke derzeit nicht hinzugefügt. Der Ansatz zeigt, wie aus stochastischen Verteilungen Strecken und somit Testfälle ableitbar sind. Dabei benötigt der Ansatz keine Informationen über den Quellcode des Testobjekts, wodurch diese Anforderung erfüllt wird. Jedoch kann keine Aussage darüber getroffen werden, inwiefern die Anzahl der redundanten Testfälle minimiert wird, da derzeit nur ein Einflussparameter einbezogen wurde und nicht beschrieben wird, wie die Kombination von verschiedenen Verteilungsfunktionen betrachtet wird.

### 8.2.3 Bewertung der Ansätze und Ableitung des Forschungsbedarfs

Die vorgestellten Beschreibungen für Szenarien zeigen, dass die Szenarien auf verschiedenen Ebenen beschrieben werden und damit einen strukturierten Aufbau haben. Der Ansatz im modularen virtuellen Testbau orientiert sich daher am Aufbau der vorgestellten Ansätze, da diese die Anforderung an eine einheitliche Struktur für Szenarien zum Teil erfüllen. Die weitere Betrachtung der Ansätze zeigt jedoch, dass derzeit keine Beschreibungsform alle gestellten Anforderungen erfüllt. Die Reproduzierbarkeit aus Sicht des Testobjekts wird in keiner Beschreibung explizit betrachtet. Einheitliche und standardisierte Strukturen sind zum Teil durch die Verwendung des *OpenDrive*-Formats zu finden. Dieses beschreibt jedoch nur den stationären Teil des Szenarios. Für die Beschreibung der beweglichen Elemente beziehungsweise Akteure ist derzeit kein standardisierter Ansatz vorhanden. Daher wird im folgenden Kapitel ein Ansatz vorgestellt, der das Potential bietet, die Anforderungen zu erfüllen und reproduzierbare Szenarien aus globaler Sicht und aus der Sicht des Testobjekts zu ermöglichen. Weiterhin soll der Ansatz eine hierarchische Struktur haben, um Abhängigkeiten zwischen verschiedenen Abstraktionsebenen abbilden zu können.

Alle vorgestellten Ansätze zur Testfallableitung erfüllen die Anforderung der Ableitung von Testfällen ohne Informationen über den Quellcode. Jedoch zeigt keiner der vorgestellten Ansätze die Möglichkeit, die Anzahl der redundanten Testfälle, d.h. in der vorliegenden Arbeit die Testfälle mit gleichen Diskretisierungsstufen der Parameter, zu minimieren. Daher wird im Testbaukasten die Methode der kombinatorischen Testfallableitung eingesetzt, welche in Kapitel 8.4.3 beschrieben wird. Die Methode bietet das Potential die Anzahl der redundanten Testfälle zu minimieren. Der Einsatz der Methode wird im Fallbeispiel des Engstellenassistenten in Kapitel 12 evaluiert.

### 8.3 Ein generisches 4-Ebenen-Modell für die Beschreibung von Szenarien

Dieses Kapitel präsentiert ein generisches Modell zur Beschreibung von Szenarien für automatisierte Fahrfunktionen. Das Modell beschreibt sowohl logische als auch konkrete Szenarien, wie sie in Kapitel 2.3.4 definiert wurden. Dabei basiert das Modell auf den vier hierarchisch aufgebauten Ebenen *Basisstreckennetzwerk*, *situationsspezifische Anpassungen am Basisstreckennetzwerk*, *Beschreibung und Regelung der Akteure* und *Umweltbedingungen*. Die hierarchisch höher angeordneten Ebenen können die untergeordneten Ebenen manipulieren. Ein Szenario wird auf Basis der untersten Ebene und einer Auswahl der höher liegenden Ebenen beschrieben. Abbildung 8.1 zeigt schematisch die vier Ebenen des Modells, die im Folgenden erläutert werden.

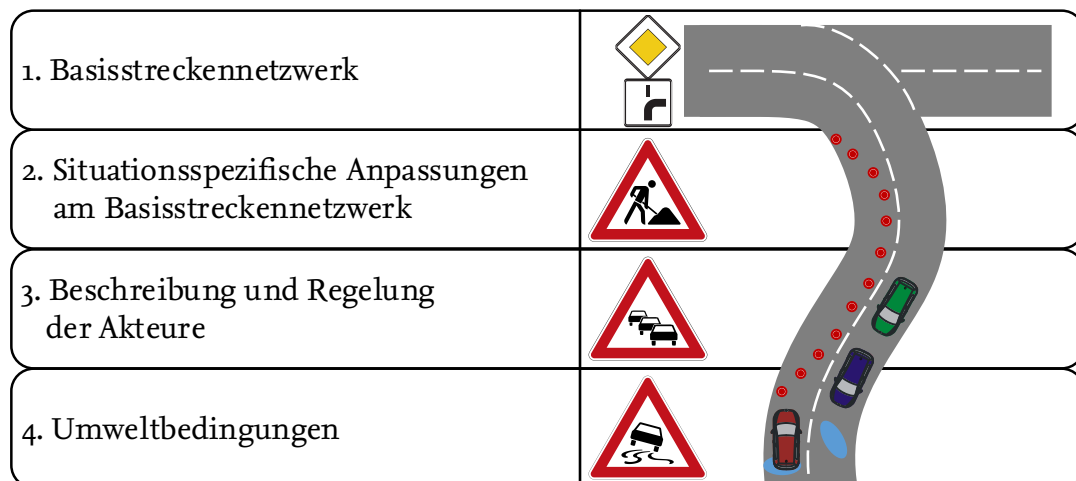


Abbildung 8.1: Aufbau des 4-Ebenen-Modells

Das 4-Ebenen-Modell und die systematische Testfallableitung wurde bereits in Schuldts u. a. (2013)<sup>5</sup> vorgestellt. Das Konzept zur Beschreibung und Regelung der Akteure wurde in Schuldts u. a. (2016)<sup>6</sup> präsentiert.

<sup>5</sup>Die Idee des 4-Ebenen-Modells wurde vom Autor der vorliegenden Arbeit erbracht. Die Koautoren haben durch intensive Reviews und Diskussionen zur Reife und Detaillierung des Modells beigetragen.

<sup>6</sup>siehe oben.

### 8.3.1 Basisstraßennetzwerk

Auf der ersten Ebene wird das Basisstraßennetzwerk definiert. Dafür werden sowohl die Geometrie als auch die Topologie der Straßen auf submikroskopischer Ebene spezifiziert. Für die Beschreibung der Geometrie werden Basiselemente der Straßenplanung, wie Geraden, Kurven und Klothoiden, verwendet. Geraden werden durch eine Länge beschrieben. Kurven werden durch eine konstante Krümmung sowie eine Länge definiert. Als Übergangselement zwischen Geraden und Kurven oder Kurven mit verschiedenen Krümmungen werden Klothoiden verwendet, die die Krümmung des Vorgängerelements linear an die Krümmung des Nachfolgeelements anpassen. Damit besteht die Möglichkeit, Straßen zu definieren, die über einen stetigen Krümmungsverlauf verfügen. Weiterhin wird das laterale und longitudinale Neigungsprofil der Straße definiert.

Die Werte zur Parametrisierung der Geometrien werden aus dem Modul der Auswahl und Analyse von Einflussparametern übernommen. Sollten diese Werte nicht spezifiziert sein, werden geltende Normen und Richtlinien für die Erstellung von Autobahnen (FGSV, 2008), Landstraßen (FGSV, 2013) und Stadtstraßen (FGSV, 2006) verwendet. Die Tabelle 8.1 zeigt einige exemplarische Werte aus den Richtlinien für die Erstellung von Autobahnen in Deutschland. Die Tabelle präsentiert die verschiedenen Entwurfsklassen für die Autobahn

Entwurfsklasse	EKA 1A	EKA 1B	EKA 2	EKA 3
Netzfunktion	Fernautobahn	Überregionalautobahn	Autobahnähnliche Straße	Stadtautobahn
Höchstlänge der Geraden L in m	2000			-
Kurvenmindestradius R in m	900	720	470	280
Klothoidenmindestparameter A in m	300	240	160	90

**Tabelle 8.1:** Auszug aus den Entwurfselemente für Autobahnen nach den Richtlinien für die Anlage von Autobahnen (FGSV, 2008). EKA: Entwurfsklasse Autobahn

und die Werte für die maximal erlaubte Länge von Geraden, für den Kurvenmindestradius sowie für den Klothoidenmindestparameter, welcher mit der Formel  $A = \sqrt{R \cdot L}$  berechnet wird.  $R$  bezeichnet den Unterschied zwischen dem Start- und Endradius der Klothoide und  $L$  beschreibt die Länge der Klothoide. Sollte eine Straße nicht mit den oben genannten Basiselementen darstellbar sein, ist es möglich den Verlauf der Straße mittels verschiedener Splines zu beschreiben. Damit wird es dem Anwender des Testbaukastens ermöglicht, nicht normgerechte Straßenverläufe zu erstellen. Dies beinhaltet auch Straßen mit Krümmungssprüngen.

Nachdem die Geometrie der Straße definiert wurde, wird im nächsten Schritt die Topologie der Straße beschrieben. Dafür wird die Fahrstreifenzahl, die Fahrstreifenbreite sowie die Markierung der Fahrstreifen definiert. Die Markierungen werden spezifiziert durch die Breite, Farbe und den Strichtyp. Diese Werte werden ebenfalls aus der Auswahl und Analyse der Einflussparameter übernommen. Sollten die Werte nicht spezifiziert sein, werden die geltenden Normen und Richtlinien für die Definition der Werte verwendet. In den Normen

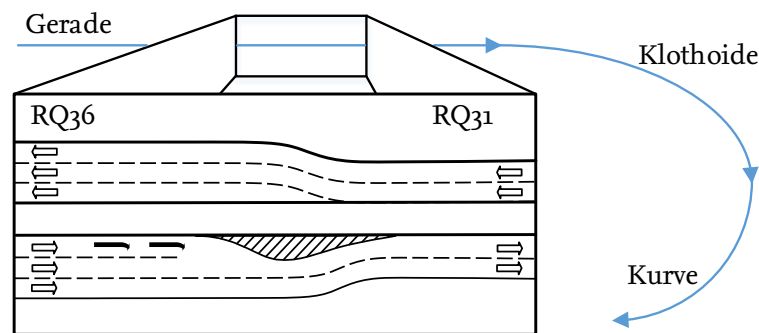


und Richtlinien werden weiterhin unterschiedliche Regelquerschnitte, wie in Kapitel 2.4 eingeführt, für verschiedene Straßentypen beschrieben. Diese definieren die Anzahl der Fahrstreifen sowie deren Breite in Abhängigkeit vom Verkehrsfluss. Tabelle 8.2 gibt einen Überblick über die Regelquerschnitte auf deutschen Autobahnen.

Name	Böschung in m	Seitenstreifen in m	Randstreifen in m	Fahrstreifen in m				Randstreifen in m	Mittelstreifen in m
				1.	2.	3.	4.		
RQ 25	1,5	2,0	0,5	3,5	3,25			0,5	2,5
RQ 28	1,5	2,5	0,75	3,5	3,5			0,75	3,0
RQ 31	1,5	3,0	0,75	3,75	3,75			0,75	4,0
RQ 36	1,5	2,5	0,5	3,75	3,5	3,5		0,75	4,0
RQ 43,5	1,5	2,5	0,5	3,75	3,75	3,5	3,5	0,75	4,0

**Tabelle 8.2:** Übersicht der Regelquerschnitte (RQ) für Autobahnen nach den Richtlinien für die Anlage von Autobahnen (FGSV, 2008)

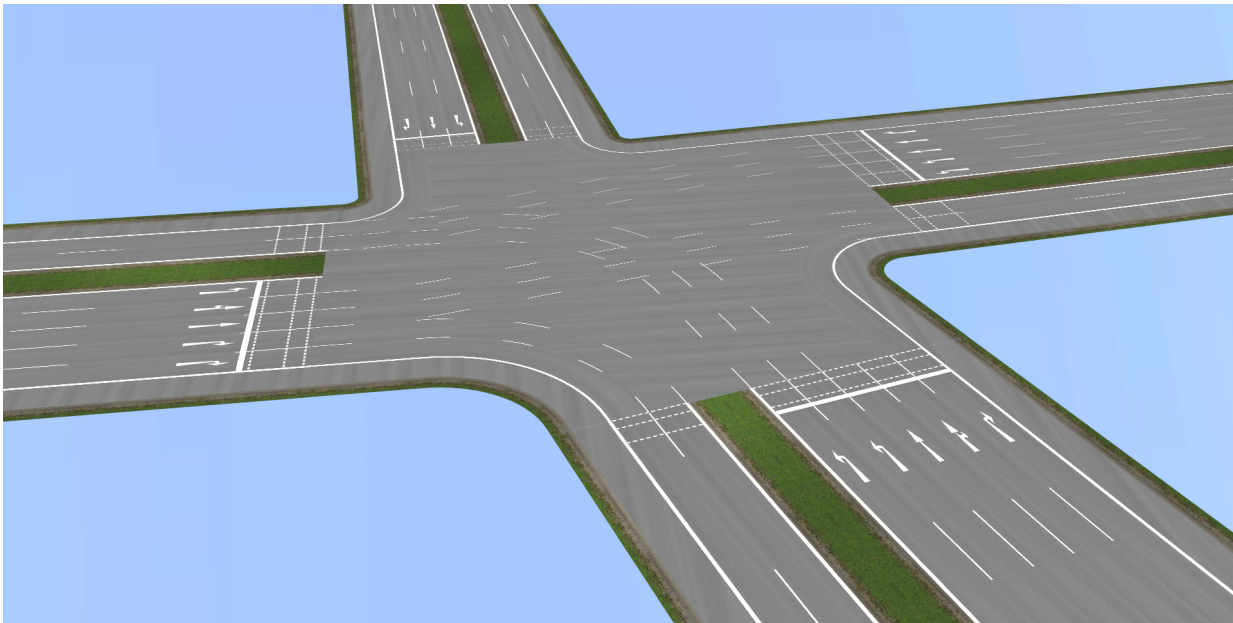
Abbildung 8.2 zeigt den Verlauf einer Straße basierend auf Geraden, Kurven und Klothoiden mit zwei Regelquerschnitten für Autobahnen. Die Abbildung zeigt ebenfalls den Übergangsbereich zwischen den beiden Regelquerschnitten, der in den Normen und Richtlinien (FGSV, 2008) definiert ist.



**Abbildung 8.2:** Beispiel einer Basisstrecke bestehend aus einer Geraden, Klothoiden und einer Kurve sowie einem Übergang von Regelquerschnitt (RQ) 31 und 36 auf der Geraden

Weiterhin besteht die Möglichkeit, Kreuzungen im Basisstraßennetzwerk zu beschreiben. Hierfür werden ebenfalls die Normen und Richtlinien, wie die Richtlinie zur Anlage von Stadtstraßen (FGSV, 2006), herangezogen, falls die Kreuzung in der Auswahl und Analyse der Einflussparameter nicht weiter spezifiziert wurde. Um Kreuzungen zu definieren, muss die Anzahl der ein- und ausgehenden Straßen sowie die Anzahl der verknüpfenden Fahrstreifen innerhalb der Kreuzung mit ihren Breiten und Markierungen definiert werden. Abbildung 8.3 zeigt ein Szenario, welches auf der ersten Ebene des Modells zur Beschreibung von Szenarien definiert und erstellt wurde. Dargestellt ist ein Kreuzungsszenario mit vier ein- und ausgehenden Straßen.

Marek (2014) beschreibt in seiner Arbeit ein Framework für die systematische Erstellung von stationären Straßennetzwerken auf der ersten Ebene. In dem Framework wurden Softwarekomponenten implementiert, die Teilaufgaben zur Erstellung eines Straßennetzwerks



**Abbildung 8.3:** Beispiel eines Kreuzungsszenarios auf der ersten Ebene<sup>7</sup>

übernehmen. Es werden Komponenten für die Linienführung der Straße, für die Generierung der Straßentopologie anhand von Regelquerschnitten, für die Generierung von normgerechten Übergängen zwischen zwei verschiedenen Regelquerschnitten und für die automatische Erstellung von Kreuzungen vorgestellt. Mittels des Frameworks und den in der Arbeit vorgestellten Komponenten ist es möglich, umfangreiche Straßennetzwerke auf der ersten Ebene des 4-Ebenen-Modells zu erstellen.

### 8.3.2 Situationsspezifische Anpassungen am Basisstraßennetzwerk

Auf der zweiten Ebene des 4-Ebenen-Modells werden situationsspezifische Anpassungen am Basisstreckennetzwerk vorgenommen, die für spezielle Anwendungen oder automatisierte Fahrfunktionen benötigt werden. Diese Anpassungen sind zum Beispiel das Hinzufügen der Randbebauungen der Straße oder andere stationäre Elemente im Straßennetzwerk, wie Schläglöcher oder Bodenschwelle, die das Profil des Basisstreckennetzwerks verändern. Für den Test mit verdeckten Elementen besteht die Möglichkeit, auf dieser Ebene eine städtische Umgebung bestehend aus Häusern, Straßenlaternen und Lichtsignalanlagen zum Basisstreckennetz hinzuzufügen. Abbildung 8.4 zeigt eine Anpassung des Kreuzungsszenarios mit einer Erweiterung auf der zweiten Ebene durch eine städtische Szenerie. Es wurden Häuser, Straßenlaternen, Verkehrszeichen und Lichtsignalanlagen hinzugefügt.

Auf einer Autobahn kann es für einen Test von Bedeutung sein, Elemente der Randbebauung, wie Verkehrszeichenbrücken, Verkehrszeichen, Leitplanken oder Schallschutzmauern, dem Szenario hinzuzufügen. Eine weitere Anpassung des Straßennetzwerks ist die Erstellung von weiteren Straßen basierend auf dem bisher existierenden Straßennetzwerk. Diese Möglichkeit der Anpassung wird beispielsweise bei der Definition und Erstellung von Szenarien mit Arbeitsstellen benötigt. Um den Verkehr in der Arbeitsstelle zu leiten, werden zusätzliche Fahrstreifenmarkierungen mit verschiedenen Farben sowie stationäre Leitelemente zum

<sup>7</sup>Das Bild wurde mit der Simulationssoftware *Virtual Test Drive* der Firma *Vires* erstellt.





**Abbildung 8.4:** Erweiterung des Kreuzungsszenarios auf der zweiten Ebene durch eine städtische Szenerie<sup>8</sup>

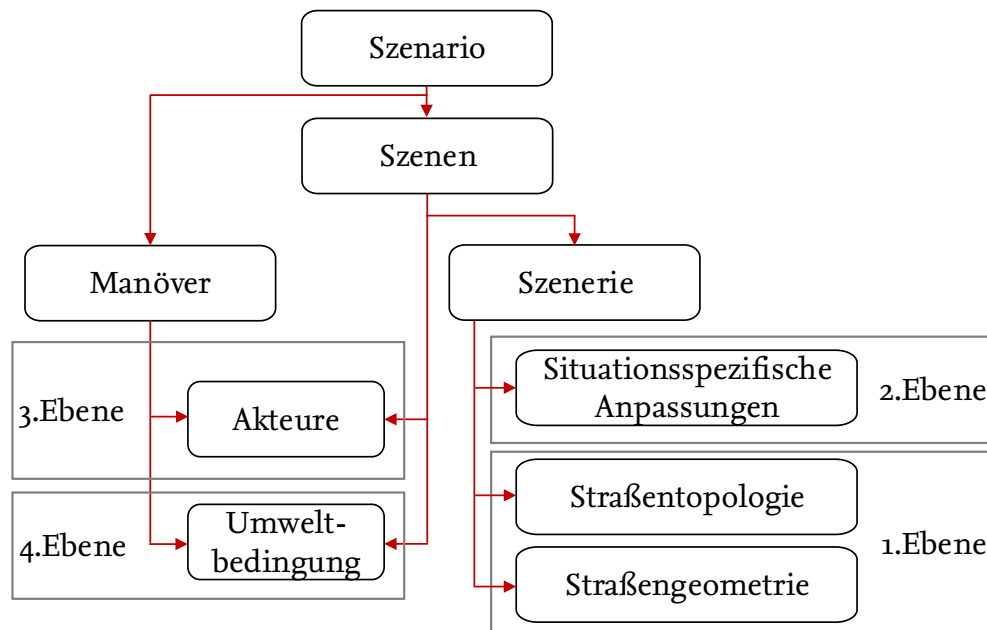
Basisstreckennetzwerk hinzugefügt. Das Basisstreckennetzwerk auf der ersten Ebene liegt weiterhin unter der angepassten Strecke, wodurch die Geometrien weiterhin existieren und sichtbar sind. Damit können die Markierungen der ursprünglichen Fahrstreifen als inkonsistente Informationen für das Testobjekt, beispielsweise für einen Fahrstreifenhalteassistenten nach ISO 11270 (2014), verwendet werden.

### 8.3.3 Beschreibung und Regelung der Akteure

Auf der dritten Ebene wird die Anzahl der beweglichen und unbeweglichen Akteure sowie deren Verhalten definiert. Die Akteure sind unter anderem Fahrzeuge, Fußgänger oder Tiere, die sich auf oder neben der Straße bewegen oder die Straße überqueren. Dadurch besteht auch die Möglichkeit, eine vom Wind angetriebene Tüte zu modellieren. Weiterhin wird die Steuerung der Lichtsignalanlagen und Wechselverkehrszeichen auf dieser Ebene definiert. Dies widerspricht der Definition, dass diese Elemente zur Szenerie der Szene gehören und damit stationär sind. Allerdings wird auf dieser Ebene nur die Steuerung der Lichtsignalanlagen und Wechselverkehrszeichen definiert. Dies entspricht dem variablen Teil der Elemente. Das physikalische Objekt und damit der stationäre Teil des Elements werden auf der zweiten Ebene definiert und gehören damit zur Szenerie des Szenarios.

Abbildung 8.5 zeigt den Aufbau eines Szenarios durch das 4-Ebenen-Modell. Ein Szenario besteht, wie in Kapitel 2.3.4 definiert, aus Szenen und Aktionen & Ereignissen, die hier zu Manövern zusammengefasst werden. Die Szenen unterteilen sich in die Szenerie, die Akteure und die Umweltbedingungen. Das Zielelement der Manöver sind zum einen die Akteure, wodurch die Möglichkeit besteht, das Verhalten der Akteure gezielt zu manipulieren. Dies wird zum Beispiel für den Test von Assistenzsystemen benötigt, da dort erst die Bedingungen für die Aktivierung des Systems erfüllen muss, damit das System getestet werden kann. Eine

<sup>8</sup>Das Bild wurde mit der Simulationssoftware *Virtual Test Drive* der Firma *Vires* erstellt.

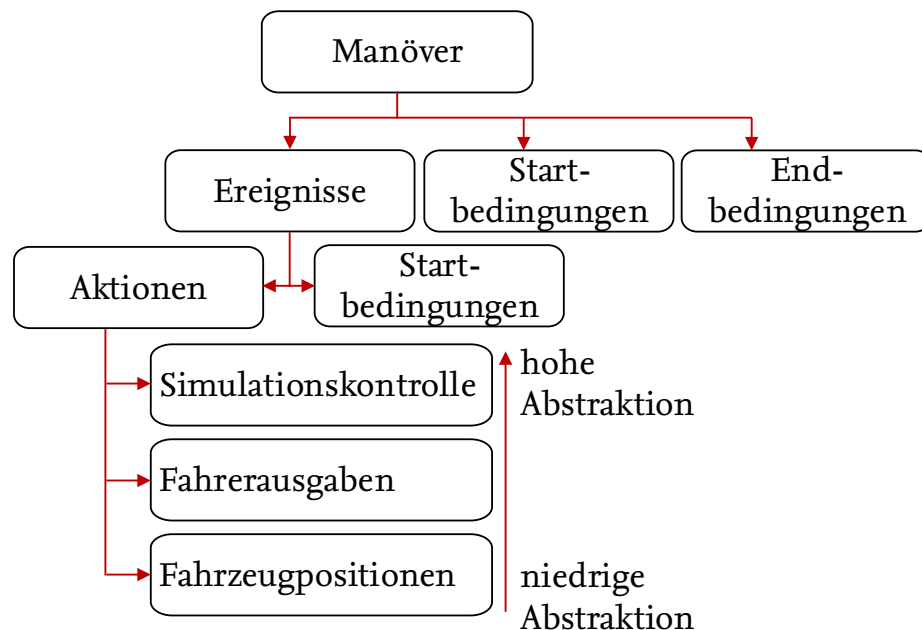


**Abbildung 8.5:** Aufbau eines Szenarios durch das 4-Ebenen-Modell

Aktivierungsbedingung für den Fahrstreifenhalteassistenten wäre beispielsweise das Verlassen des eigenen Fahrstreifens. Zum anderen werden die Manöver auf die Umweltbedingungen angewendet. Damit wird die Möglichkeit geschaffen, die Umweltbedingungen, abweichend von der grundsätzlichen Definition auf der vierten Ebene, orts- beziehungsweise zeitgebunden zu verändern. Weiterhin ist durch die Regelung der Umweltbedingungen durch die Manöver eine Koppelung zwischen den Umweltbedingungen und dem Verhalten der Akteure möglich. Dies wird für die lokale Reproduzierbarkeit benötigt, da für die Erfüllung dieser Anforderung auch die Umweltbedingungen an das Verhalten der Akteure anpassbar sein müssen.

Im Folgenden wird der Aufbau der Manöver und die Regelung der Akteure durch die Manöver beschrieben. Jedes Manöver besitzt die interne Struktur, welche in Abbildung 8.6 dargestellt ist. Ein Manöver besteht aus Ereignissen und mindestens einer oder mehrerer Start- und Endbedingungen. Die Startbedingungen sind beispielsweise der relative Abstand oder die relative Geschwindigkeit zu anderen Elementen im Szenario oder eine spezielle Charakteristik der Straße, wie eine bestimmte Kurvenkrümmung. Damit ist es möglich, die Manöver an definierten Orten auszuführen. Sollte eine der Startbedingungen erfüllt sein, wird das Manöver auf das Zielelement angewendet. Sobald eine der Endbedingungen erfüllt ist, wird das Manöver beendet und gegebenenfalls ein Folgemanöver ausgeführt.

Die Ereignisse werden verwendet, um das gewünschte orts- und/oder zeitgebundene Verhalten der Akteure zu erreichen. Jedes Ereignis hat wiederum eine oder mehrere Startbedingungen. Sobald eine Startbedingung erfüllt ist, werden die definierten Aktionen ausgeführt. Die Bedingungen für den Start der Ereignisse können an das Verhalten der Testobjekte gekoppelt werden, um die Ereignisse abhängig vom Verhalten des Testobjekts auszulösen. Dies wird für die lokale Reproduzierbarkeit aus der Sicht des Testobjekts benötigt. Für die Regelung der Akteure werden während der Durchführung des Szenarios in jedem Simulationsschritt die Informationen über das Testobjekt und die übrigen Elemente im Szenario verwendet. Je



**Abbildung 8.6:** Aufbau eines Manövers mit Ereignissen und Aktionen

nachdem wie sich das Testobjekt verhält, werden die Aktionen des Manövers angepasst, um eine lokale Reproduzierbarkeit der Szenarien aus der Sicht des Testobjekts zu erreichen.

Das folgende Beispiel nach Schuldt (2014) und Symkenberg (2014, S.35) verdeutlicht die Anpassung der Aktionen der übrigen Verkehrsteilnehmer an das Verhalten des Testobjekts. Das Testobjekt, in diesem Fall ein Fahrzeug mit manuellem Fahrer, fährt mit unterschiedlichen Geschwindigkeiten auf eine Kreuzung zu. Ein anderer beweglicher Verkehrsteilnehmer soll das Testobjekt auf der Kreuzung mit einem definierten Abstand von rechts kreuzen. Dies soll unabhängig von der Geschwindigkeit des Testobjekts möglich sein. Für die Realisierung wird durch Aktionen die Geschwindigkeit des beweglichen Akteurs in jedem Zeitschritt der Simulation an das Verhalten des Testobjekts angepasst, um die Zielszene zu erreichen. Damit ist das Erreichen der Zielszene unabhängig von der Geschwindigkeit des Testobjekts möglich. Symkenberg (2014) gibt einen weiteren Überblick über die Umsetzung der Manöver, Ereignisse sowie Aktionen für das Erreichen der lokalen Reproduzierbarkeit.

Die Aktionen werden auf verschiedenen Ebenen der Simulation durchgeführt. Wenn die Simulationsumgebung die Möglichkeit der direkten Ausführung der gewünschten Aktion anbietet, wird diese Schnittstelle für die Umsetzung der Aktion verwendet. In der Abbildung 8.6 wird dies durch den Block der Simulationskontrolle dargestellt. Wenn für ein bewegliches Element die Aktion eines Fahrstreifenwechsels verfügbar ist, wird diese Schnittstelle verwendet, um die Aktion umzusetzen. Andere mögliche Aktionen sind unter anderem das Erstellen von Elementen in der Simulation, das Ändern der Geschwindigkeit oder das Abfahren von vorgegebenen Trajektorien. Wenn eine solche Schnittstelle nicht existiert, werden die Aktionen auf einer niedrigeren Abstraktionsebene durchgeführt. Dies ist beispielsweise für den Fahrstreifenwechsel die Vorgabe von Fahrerausgaben, wie Lenkwinkel oder Gaspedalstellung, um das bewegliche Element zu steuern. Damit wird die Möglichkeit geschaffen, den Fahrstreifenwechsel detaillierter durch eine Trajektorie vorzugeben, wodurch eine Vielzahl von verschiedenen Fahrstreifenwechseln simuliert und getestet werden kann (Symkenberg,



2014). Wenn die Aktion nicht mit Fahrerausgaben ausführbar ist, wird die Aktion durch das Setzen des Zustands des beweglichen Elements in jedem Zeitschritt realisiert. Damit werden Aktionen umgesetzt, die nicht durch verfügbare Modelle abbildbar sind.

Die Umsetzung und Validierung des Konzepts zur Regelung der beweglichen Elemente werden von Symkenberg (2014) vorgestellt. In der Arbeit wird anhand eines Kreuzungsszenarios gezeigt, dass eine lokale Reproduzierbarkeit in Szenarien mit Hilfe dieser Beschreibung der Akteure realisierbar ist. Abbildung 8.7 zeigt das bereits vorgestellte Kreuzungsszenario. Dieses wurde auf der dritten Ebene des Modells um die Akteure erweitert.



**Abbildung 8.7:** Erweiterung des Kreuzungsszenarios auf der dritten Ebene durch Akteure<sup>9</sup>

### 8.3.4 Umweltbedingungen

Auf der vierten Ebene des Modells werden die grundsätzlichen Umweltbedingungen des Szenarios definiert. Eine Änderung der Umweltbedingungen während der Durchführung des Szenarios wird durch die Manöver möglich, welche im vorherigen Kapitel vorgestellt wurden. Die Umweltbedingung des Szenarios setzen sich im 4-Ebenen-Modell aus dem Wetter und der Tageszeit zusammen. Mögliche Zustände für die Tageszeit sind unter anderem die Morgendämmerung, der Tag, die Abenddämmerung oder die Nacht. Dafür ist es erforderlich zum Beispiel den Sonnenstand zu definieren. Damit besteht die Möglichkeit, automatisierte Fahrfunktionen unter verschiedenen Lichtbedingungen zu testen und beispielsweise eine gezielte Blendung der Wahrnehmung zu erreichen. Dies ist besonders für videobasierte Systeme relevant. Eine weitere Variationsmöglichkeit der Umweltbedingung ist das Wetter. Hier sind zum Beispiel die Zustände: sonnig, wolkig, regnerisch und verschneit möglich.

Durch die Definition der Umweltbedingungen werden die unteren Ebenen des 4-Ebenen-Modells beeinflusst, da beispielsweise ein regnerisches Wetter einen Einfluss auf den Reibwert und auf die Reflexionen der Straßen hat. Dadurch besteht zum Beispiel die Möglichkeit, ein regennasses Basisstreckennetzwerk zu erzeugen. Zusätzlich aktivieren Straßenlaternen und

<sup>9</sup>Das Bild wurde mit der Simulationssoftware *Virtual Test Drive* der Firma *Vires* erstellt.

bewegliche Verkehrsteilnehmer basierend auf der Tageszeit das Licht. Dies führt wiederum zu veränderten Sichtverhältnissen. Abbildung 8.8 zeigt das vollständig beschriebene Szenario auf allen vier Ebenen. Auf der vierten Ebene wurde das Wetter auf regnerisch und die Tageszeit auf Nacht definiert. In der Abbildung ist zu sehen, dass die Verkehrsteilnehmer das Abblendlicht aktiviert haben und sich somit Reflexionen auf der nassen Fahrbahn ergeben.



**Abbildung 8.8:** Erweiterung des Kreuzungsszenarios auf der vierten Ebene durch Umweltbedingungen<sup>10</sup>

### 8.3.5 Zusammenfassung

Das 4-Ebenen-Modell bietet aufgrund des Aufbaus und der strukturierten Abfolge zur Definition von Szenarien die Möglichkeit, systematisch submikroskopische Szenarien aufzubauen. Die Szenarien können auf den verschiedenen Ebenen variiert werden, um so neue Testfälle abzuleiten. Durch die Beschreibung und Regelung der Akteure auf der dritten Ebene besteht die Möglichkeit, aus Sicht des Testobjektes reproduzierbare Szenarien zu erzeugen. Dieser Aspekt wurde in den bisherigen Beschreibungen von Szenarien für Verkehrssimulationen nicht beachtet. Die Ideen und Ansätze zur Beschreibung und Regelung der Akteure wurden in mehreren Workshops (Schuldt u. a., 2015a) in den Entwurf des neuen Standards *OpenScenario*<sup>11</sup> zur Beschreibung von Szenarien eingebracht. Damit konnten Teile des 4-Ebenen-Modells zur Definition des neuen Standards beitragen. Die praktische Anwendung des Modells wird anhand des Engstellenassistenten in Kapitel 12 gezeigt.

Mit Hilfe des 4-Ebenen-Modells und der bereits beschriebenen Auswahl und Analyse von Einflussparametern wird die Definition von verschiedenen Arten von Strecken möglich. Diese können in einem Diagramm unterteilt werden, welches durch zwei ordinale Skalen aufgespannt wird. Auf der ersten Achse wird die Repräsentativität der erstellten Strecke gegenüber allen Strecken einer Domäne aufgetragen. Auf der zweiten Achse wird die Repräsentativität der erstellten Strecke gegenüber einer realen Strecke verglichen. Die Repräsentativität einer

<sup>10</sup>Das Bild wurde mit der Simulationssoftware *Virtual Test Drive* der Firma *Vires* erstellt.

<sup>11</sup><http://openscenario.org>, abgerufen: 29.02.2016

Strecke beschreibt in der vorliegenden Arbeit die Übereinstimmung von Charakteristika der beiden Strecken. Charakteristika sind beispielsweise die statistische Verteilung von Kurvenkrümmungen, Fahrstreifenanzahl oder Fahrstreifenbreiten. Damit orientiert sich der Begriff an der Definition der Statistik. Dort ist eine Stichprobe repräsentativ, wenn „sie die innere Struktur und die Charakteristika einer statistischen Grundgesamtheit adäquat abbildet“ (Eckstein, 2014, S.16).

Nachgebaute Strecken oder Straßennetzwerke mit einem realen Vorbild haben den Vorteil, dass Vorbilder für diese Strecken existieren und somit die synthetischen Strecken eine hohe Repräsentativität für diese Vorbildstrecken besitzen. Mit Tests auf dieser Strecke kann gezeigt werden, dass das Testobjekt oder die automatisierte Fahrfunktion die Strecke mit realem Vorbild in der Simulation beherrscht. Durch den Test wird jedoch nur bedingt gezeigt, dass andere existierende Strecken beherrscht werden, da diese eventuell andere Charakteristika aufweisen. Die nachgebauten Strecken haben somit eine geringe Repräsentativität gegenüber anderen Strecken.

Ein komplett erdachtes Straßennetzwerk hat den Vorteil, dass dieses mit wenigen Informationen und automatisiert erzeugbar ist. Jedoch besteht die Gefahr, dass dieses Straßennetzwerk keinen Bezug zur Realität hat und dadurch die Repräsentativität gegenüber allen Strecken gering ist. Trotzdem kann der Test auf diesen Strecken nützlich sein, da zufällig erzeugte Strecken getestet werden, die eventuell bei einer systematischen Ableitung nicht getestet werden.

Strecken und Straßennetzwerke, die nach geltenden Normen und Richtlinien modelliert werden, stellen eine weitere Art von möglichen Strecken dar. Diese haben auf beiden Achsen des Diagramms eine höhere Repräsentativität als frei erfundene Strecken, da sich die Strecken an geltende Grenzwerte halten. Die Repräsentativität hängt jedoch stark davon ab, ob die realen Strecken im Einsatzgebiet des Testobjekts den Normen und Richtlinien entsprechen. Mit Tests auf diesen Strecken wird gezeigt, dass die automatisierte Fahrfunktion auf normgerechten Strecken funktioniert. Damit ist das getestete Verhalten der automatisierte Fahrfunktion nicht direkt an eine ausgewählte Strecke gekoppelt.

Das normgerechte Straßennetz kann in der Repräsentativität durch die statistische Auswertung von real existierenden Straßen verbessert werden. Durch die statistische Auswertung wird gezeigt, welche Verteilungen von Kurvenkrümmungen, Fahrstreifenbreiten oder Höhenverläufe im Einsatzgebiet des Testobjekts vorhanden sind. Diese Verteilungen können auf die erzeugte Strecke übertragen werden, um die Repräsentativität zu erhöhen. Vorteil dieses Vorgehens ist eine erhöhte Repräsentativität der erzeugten Strecke und eine mögliche Bestätigung der Normen und Richtlinien. Diese Arten von Straßennetzwerken könnten das Potential bieten, als *Referenzstrecken* auf den ersten beiden Ebenen des Modells zu dienen, da sie durch die Auswertung und Analyse von realen Straßen mit der Kombination von Normen und Richtlinien als Referenz für eine Vielzahl von Strecken stehen könnten. Dies ist jedoch nur eine Vermutung, die in dieser Arbeit nicht nachgewiesen wird. Abbildung 8.9 zeigt die Einteilung der Strecken im beschriebenen Diagramm mit den beiden ordinalen Skalen der Repräsentativität von Strecken.



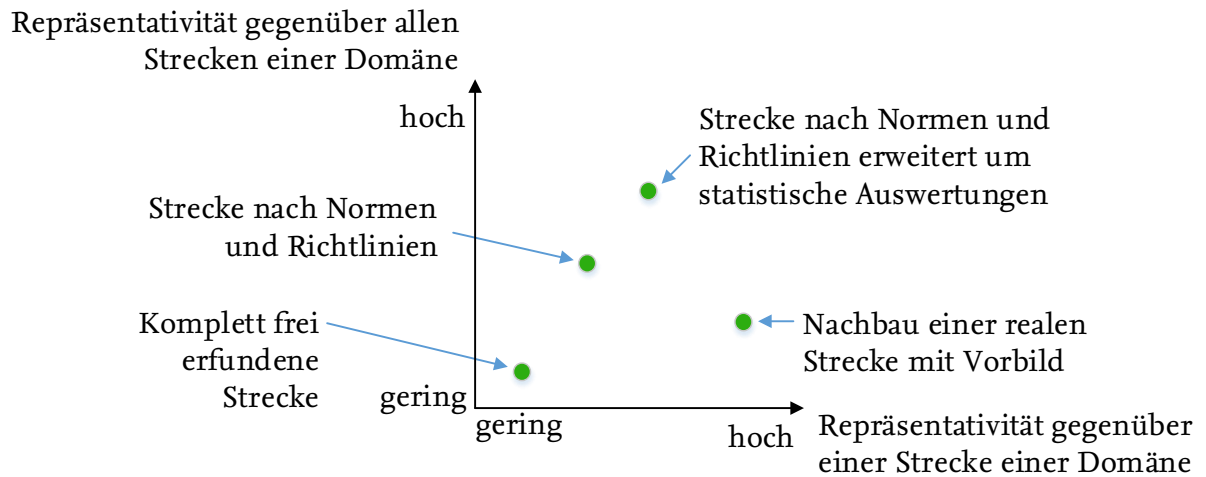


Abbildung 8.9: Repräsentativität von verschiedenen Strecken

## 8.4 Systematische Testfallableitung

Die systematische Testfallableitung erstellt auf Basis von logischen Szenarien konkrete Testfälle. Dies bedeutet, dass für jeden Testfall die Szenarioparameter festgelegt werden. Auf jeder Ebene des 4-Ebenen-Modells werden je nach Detaillierungsgrad der Beschreibung verschiedenste Parameter variiert. Ein Szenario mit fünf Einflussparametern und sechs Diskretisierungsstufen für jeden Parameter generiert bereits  $6^5 = 7.776$  Testfälle. Bei jedem hinzugefügten Parameter oder jeder zusätzlichen Diskretisierungsstufe steigt die Anzahl der durchführbaren Testfälle progressiv an. Aufgrund von Zeit- und Kostenlimitation ist es nicht möglich, alle theoretisch erstellbaren Testfälle durchzuführen (Sommerville, 2006, S.539). Dies ist einer der Gründe für den Bedarf neuer Methoden, um die Anzahl der Testfälle bei einer definierten Testabdeckung zu reduzieren.

Die Herausforderung des progressiven Wachstums der Anzahl der möglichen Testfälle ist ein bekanntes Problem im Forschungsbereich des Software-Testens. Eine Möglichkeit zur Testfallableitung ist nach Nie und Leung (2011) die Verwendung von kombinatorischen Algorithmen, um das progressive Wachstum der Anzahl von Testfällen während der Testfallableitung zu verhindern. Diese Algorithmen reduzieren die Anzahl der Testfälle auf der Basis von kombinatorischen Überlegungen. Kuhn u. a. (2004) zeigen in einer Fallstudie, dass im Schnitt 70% - 97% der Fehler gefunden werden, wenn jedes Paar der Diskretisierungsstufen in mindestens einem Testfall getestet wird. In der Fallstudie wurden Systeme verschiedener Domänen getestet.

Der Test von automatisierten Fahrfunktionen stellt das Testkonzept vor ähnliche Herausforderungen, da aufgrund der vielen Variationsmöglichkeiten die Anzahl der möglichen Testfälle progressiv wächst. Daher werden im modularen virtuellen Testbaukasten in der systematischen Testfallableitung die kombinatorischen Algorithmen für die gezielte Reduktion der Anzahl der Testfälle evaluiert. Neben der kombinatorischen Testfallableitung werden die Testmethoden der Äquivalenzklassenbildung und der Grenzwertanalyse verwendet. Die beiden Verfahren gehören zu der Gruppe der partitionierenden Testmethoden, da die Methoden die Parameter und Diskretisierungsstufen in Gruppen aufteilen. Die Methoden werden vorgelagert zu den kombinatorischen Tests verwendet.



Robinson-Mallet u. a. (2010) empfehlen ebenfalls die Verwendung der drei Testmethoden für den Test von Assistenzsystemen im Kontext zur ISO 26262 (2011). Die Methoden erbringen den Nachweis für die Erfüllung von funktionalen und nicht-funktionalen Anforderungen, der Abdeckung interner und externer Schnittstellen und der Abdeckung funktionaler Abhängigkeiten. Die Methoden eignen sich nicht für den Nachweis aller Benutzungsfälle und Randbedingungen. Robinson-Mallet u. a. (2010) empfehlen für den Nachweis aller Benutzungsfälle und Randbedingungen die Verwendung von statistischen Tests (modellbasierten Test) und Sequenz-Enumerationen (Methode der formalen Analyse). Da der Testbaukasten aktuell weder Methoden der formalen Analyse nutzt, noch Modelle für das System für einen modellbasierten Test erstellt, können die empfohlenen Methoden für den Nachweis aller Benutzungsfälle und Randbedingungen nicht verwendet werden. Damit erbringt der Testbaukasten mit den gewählten Testmethoden derzeit nicht den Nachweis aller Benutzungsfälle und Randbedingungen für das Testobjekt. Jedoch liegt der Fokus des Testbaukastens, wie in Kapitel 6.1 beschrieben, auf dem funktionalen Test von automatisierten Fahrfunktionen. Nach Robinson-Mallet u. a. (2010) bieten die systematischen Testverfahren der partitionierenden und kombinatorischen Testmethoden das Potential, den Nachweis für die Erfüllung von funktionalen und nicht-funktionalen Anforderungen zu erbringen. Daher wird der Einsatz der Äquivalenzklassenmethode, der Grenzwertbetrachtung und der kombinatorischen Testfallableitung im Testbaukasten evaluiert.

Die einzelnen Schritte der systematischen Testfallableitung werden im Folgenden beschrieben. Dabei werden ebenfalls die Herausforderungen der Anwendung der Testmethoden im Kontext von automatisierten Fahrfunktionen diskutiert.

### 8.4.1 Äquivalenzklassen

Das Ziel von Äquivalenzklassen ist die Generierung einer hohen Testabdeckung mit einer begrenzten Anzahl von Testfällen. Dafür werden Äquivalenzklassen durch die Gruppierung von Parameterwerten gebildet. Alle Repräsentanten einer Klasse sollen das gleiche Verhalten beim Testobjekt hervorrufen. Damit sollte durch den Test mit einem Repräsentanten jeder Klasse die komplette Klasse von Parameterwerten als getestet gelten. Liggesmeyer (2009, S.51ff) spricht dabei von funktionalen Äquivalenzklassen, da die Klassen von der Spezifikation des Systems abgeleitet werden.

Block u. a. (2014) zeigen anhand eines Beispiels, dass die Anwendung von funktionalen Äquivalenzklassen im Bezug zu automatisierten Fahrfunktionen eine große Herausforderung darstellt, da gegebenenfalls schon kleine Änderungen der Parameterwerte zu einem anderen Verhalten des Testobjekts führen und diese nicht unbedingt durch die Grenzen der Äquivalenzklassen abgebildet werden. Eine Alternative ist die Verwendung von Äquivalenzklassen zur nicht funktionalen Gruppierung von Diskretisierungsstufen. Dabei findet eine Gruppierung der Diskretisierungsstufen und Wertebereiche statt, die nicht auf Basis der Spezifikation begründet ist, sondern aus anderen Informationsquellen, wie in Kapitel 7.3 und 7.4 vorgestellt, hergeleitet wird. Diese nicht funktionalen Äquivalenzklassen werden im Testbaukasten verwendet, um die Wertebereiche beziehungsweise Diskretisierungsstufen im Schema zur Darstellung von Einflussparametern und im 4-Ebenen-Modell zu gruppieren und Testfälle abzuleiten. Dabei werden valide und invalide Klassen definiert. Während valide Klassen die Werte beinhalten, die beispielsweise innerhalb der Normen und Richtlinien liegen, bestehen invalide Klassen aus Werten, die außerhalb der Normen und Richtlinien

liegen. Automatisierte Fahrfunktionen sollten sowohl mit validen als auch mit invaliden Klassen getestet werden, um die Robustheit des Systems zu testen.

### 8.4.2 Grenzwertanalyse

Die Grenzwertanalyse beschreibt eine ergänzende Möglichkeit zur Ableitung von Testfällen aus Äquivalenzklassen. Der Unterschied zur ursprünglichen Äquivalenzklassenmethode besteht in der Strategie, einen Repräsentanten der Klasse auszuwählen. Während bei der Äquivalenzklassenmethode ein zufälliger Repräsentant aus der Klasse ausgewählt wird, werden bei der Grenzwertanalyse gezielt die Grenzwerte der Äquivalenzklassen für die Tests ausgewählt. Arnican (2009) stellt einen Überblick über mögliche Methoden der Grenzwertanalyse vor. Durch die Grenzwertanalyse ist es möglich, Repräsentanten zu finden, die eine höhere Fehlersensitivität haben als andere Repräsentanten, d.h. mit einer höheren Wahrscheinlichkeit zu einem Fehler führen (Liggesmeyer, 2009, S.196). Die Grenzwertanalyse wird als zusätzliches Mittel in der Testfallableitung verwendet.

### 8.4.3 Kombinatorische Testfallableitung

Das Ziel der kombinatorischen Testfallableitung ist die Minimierung von redundanten Testfällen, wie sie in Kapitel 8.1.2 gefordert wurde. Dazu werden gezielt die Diskretisierungsstufen variiert, um redundante Testfälle in Form von Testfällen mit gleichen Diskretisierungsstufen zu minimieren. Die verschiedenen Testabdeckungen und Algorithmen für diese kombinatorische Ableitung werden im nächsten Abschnitt beschrieben. Ein umfassender Überblick über kombinatorische Algorithmen ist in Kuhn u. a. (2013) zu finden.

#### 8.4.3.1 Testabdeckung einer kombinatorischen Testfallableitung

Die kombinatorischen Algorithmen erzeugen Testsuites, wie eine Sammlung von Testfällen bezeichnet wird, mit verschiedenen Testabdeckungen. Dabei wird davon ausgegangen, dass bereits diskrete Parameterwerte vorliegen.

Die schwächste Testabdeckung ist das Abdeckungskriterium *each-used* (Übersetzung: jeweils einmal verwendet). Diese Abdeckung wird mit einer Testsuite erreicht, bei der jeder Parameterwert in mindestens einem Testfall vorhanden ist (Grindal u. a., 2005). Eine Testsuite mit dieser Testabdeckung wird verwendet, um das Verhalten bei der Eingabe von gültigen und ungültigen Parameterwerten zu testen. Jedoch sind die Ergebnisse dieser Tests nicht repräsentativ für das Testobjekt, da nach Kuhn u. a. (2004) Fehler meist bei einer Kombination von zwei oder mehr Parameterwerten auftreten. Diese Kombinationen werden mit einer hohen Wahrscheinlichkeit nicht mit einer Testsuite getestet, die die beschriebene Testabdeckung besitzt.

Ein Abdeckungskriterium mit einer höheren Testabdeckung ist die paarweise Testabdeckung. Dieses Kriterium ist erfüllt, wenn jedes Paar von Parameterwerten in mindestens einem Testfall vorhanden ist. Durch die paarweise Testabdeckung werden Fehler mit einer höheren Wahrscheinlichkeit erkannt, da Fehler aufgedeckt werden, die auf der Kombination von zwei Parameterwerten basieren (Grindal u. a., 2005). Dadurch besteht die Möglichkeit, die Interaktion der Parameterwerte zu analysieren, da jedes Paar von Parameterwerten in mindestens einem Testfall vorhanden ist. Die Testabdeckung wird vergrößert, wenn die Testsuite anstelle der paarweisen Abdeckung eine *t-wise* (Übersetzung: *t*-weise) Abdeckung besitzt. Damit wird die Interaktion von *t* Parameterwerten getestet.

Das größte Abdeckungskriterium ist die *N-wise* (Übersetzung: N-weise) Testabdeckung. Um diese Abdeckung zu erreichen, müssen alle möglichen Kombinationen der Parameterwerte durch die Testfälle in der Testsuite abgedeckt sein (Grindal u. a., 2005). Dabei wird von einem erschöpfenden Test gesprochen, welcher jedoch aufgrund des progressiven Wachstums der Testfälle nicht für jedes Testobjekt durchführbar ist (Sommerville, 2006, S.539).

Um die Entwicklung der Größe der Testsuite bei verschiedenen Testabdeckungen zu demonstrieren, werden für das Anwendungsbeispiel der exemplarischen Fahrfunktion aus Kapitel 3.3 verschiedene Testsuites erstellt. In der Auswahl und Analyse wurden die in Tabelle 8.3 dargestellten Parameter und Diskretisierungsstufen abgeleitet.

Einflussparameter	Diskretisierungsstufen				
Fahrstreifenbreite in m	3,25	3,50	3,75		
Radius in m	-1665	-1270	-280	280	1485
Linke Seite: Markierungstyp	Durchgezogen	Unterbrochen Abstand 3m	Unterbrochen Abstand 12m		
Linke Seite: Markierungsbreite in m	0,15	0,30			
Rechte Seite: Markierungstyp	Durchgezogen	Unterbrochen Abstand 3m	Unterbrochen Abstand 12m		
Rechte Seite: Markierungsbreite m	0,15	0,30			

**Tabelle 8.3:** Übersicht der ausgewählten Einflussparameter und Diskretisierungsstufen für die exemplarische Fahrfunktion

Um die *each-used* (Übersetzung: jeweils einmal verwendet) Testabdeckung zu erreichen, werden fünf Testfälle benötigt. Tabelle 8.5 stellt dies dar.

Testfälle	Breite	Radius	Linke Seite:		Rechte Seite:	
			Markierungstyp	Markierungsbreite	Markierungstyp	Markierungsbreite
1	3,50m	-1665m	Unterbrochen 12m	0,30m	Unterbrochen 12m	0,30m
2	3,75m	-1270m	Unterbrochen 3m	0,15m	Unterbrochen 3m	0,30m
3	3,25m	-280	Durchgezogen	0,30m	Durchgezogen	0,30m
4	3,50m	280m	Unterbrochen 3m	0,15m	Unterbrochen 3m	0,30m
5	3,25m	1485m	Durchgezogen	0,15m	Unterbrochen 3m	0,30m

**Tabelle 8.4:** Testsuite mit der Testabdeckung: *each-used* (Übersetzung: jeweils einmal verwendet)

Für eine paarweise Testabdeckung werden für das Testobjekt 15 Testfälle benötigt. Dies wird in Tabelle 8.4 dargestellt.

Test-fälle	Breite	Radius	Linke Seite:		Rechte Seite:	
			Markierungs- typ	Markie- rungsbreite	Markierungs- typ	Markie- rungsbreite
1	3,25m	-1665m	Unterbrochen Abstand 12m	0,30m	Unterbrochen Abstand 12m	0,30m
2	3,50m	-1665m	Unterbrochen Abstand 3m	0,15m	Unterbrochen Abstand 3m	0,15m
3	3,75m	-1665m	Durchgezo- gen	0,30m	Durchgezo- gen	0,15m
4	3,25m	-1270m	Unterbrochen Abstand 3m	0,15m	Durchgezo- gen	0,30m
5	3,50m	-1270m	Durchgezo- gen	0,30m	Unterbrochen Abstand 12m	0,15m
6	3,75m	-1270m	Unterbrochen Abstand 12m	0,15m	Unterbrochen Abstand 3m	0,30m
7	3,25m	-280m	Durchgezo- gen	0,30m	Unterbrochen Abstand 3m	0,15m
8	3,50m	-280m	Unterbrochen Abstand 12m	0,15m	Durchgezo- gen	0,30m
9	3,75m	-280m	Unterbrochen Abstand 3m	0,30m	Unterbrochen Abstand 12m	0,30m
10	3,25m	280m	Durchgezo- gen	0,15m	Unterbrochen Abstand 3m	0,30m
11	3,50m	280m	Unterbrochen Abstand 12m	0,30m	Durchgezo- gen	0,15m
12	3,75m	280m	Unterbrochen Abstand 3m	0,15m	Unterbrochen Abstand 12m	0,30m
13	3,25m	1485m	Durchgezo- gen	0,30m	Unterbrochen Abstand 3m	0,30m
14	3,50m	1485m	Unterbrochen Abstand 12m	0,15m	Durchgezo- gen	0,15m
15	3,75m	1485m	Unterbrochen Abstand 3m	0,15m	Unterbrochen Abstand 12m	0,30m

Tabelle 8.5: Testsuite mit einer paarweisen Testabdeckung

Für eine *3-wise* (Übersetzung: 3-weise) Testabdeckung werden 54 Testfälle benötigt, für eine *4-wise* (Übersetzung: 4-weise) Testabdeckung 137 Testfälle, für eine *5-wise* (Übersetzung: 5-weise) Testabdeckung 270 Testfälle und für eine *6-wise* (Übersetzung: 6-weise) Testabdeckung, die der *N-wise* (Übersetzung: N-weise) Testabdeckung entspricht, 540 Testfälle. Das Beispiel zeigt, wie sich die Wahl der Testabdeckung auf die Größe der Testsuite auswirkt.

#### 8.4.3.2 Algorithmen für die Ableitung von kombinatorischen Testfällen

Grindal u. a. (2005) und Nie und Leung (2011) stellen jeweils eine Übersicht über mögliche Algorithmen zur kombinatorischen Testfallableitung vor. Die Algorithmen werden in

deterministische und nicht deterministische Algorithmen unterteilt, wobei deterministische Algorithmen stets die gleiche Testsuite bei vorgegebenen Eingangsparametern erstellen. Im Folgenden werden nur die deterministischen Algorithmen betrachtet, da durch die Anforderung der Reproduzierbarkeit aus Kapitel 5.1.2 immer die gleiche Testsuite bei vorgegebenen Parametern erstellt werden muss.

Die deterministischen Algorithmen werden in iterativ und instantan erstellende Algorithmen unterteilt. Instantan erstellende Algorithmen generieren die Testsuite mit einem geringen Zeitaufwand, da die Testsuite auf bereits existierenden Matrizen aufgebaut wird. Die Algorithmen sind jedoch nicht für jedes Testobjekt einsetzbar, da diese nur auf spezielle Testkonfigurationen mit einer definierten Anzahl an Parametern und Parameterstufen anwendbar sind (Grindal u. a., 2005). Die iterativ arbeitenden Algorithmen sind dagegen auf jedes Testobjekt anwendbar, da sie unabhängig von der Testkonfiguration sind. Nachteil ist jedoch, dass die Erstellung der Testsuite abhängig von der Anzahl der Parameter und Parameterwerte im Verhältnis zum instantan erstellenden Algorithmus einen höheren Zeitaufwand benötigt. Der benötigte Zeitaufwand für die Erstellung einer Testsuite mit verschiedenen Testkonfigurationen wurde in der Arbeit von Schulze (2014) untersucht. Es wurden Testsuites mit einer *3-wise* (Übersetzung: 3-weisen) Testabdeckung erstellt. Die Arbeit zeigt, dass auf dem verwendeten Rechner bei Testkonfiguration mit 20 Parametern und jeweils 4 Stufen je Parameter die Berechnungszeit bei ca. 25 Sekunden liegt. Bei einer Erhöhung der Parameter zeigt sich ein exponentielles Wachstum der Berechnungszeit. Für eine Testkonfiguration mit 40 Parametern und jeweils 4 Stufen je Parametern wird eine Berechnungszeit von ca. 275 Sekunden angegeben.

Die Testfälle können bei den deterministisch iterativ erstellenden Algorithmen anhand von Parametern oder bereits existierenden Testfällen abgeleitet werden. Der Algorithmus *IPOG* (*In Parameter Order General*), welcher von Lei u. a. (2008) vorgestellt wird und von Schulze (2014) für die oben beschriebene Bewertung der Berechnungszeit verwendet wurde, ist ein Beispiel für einen iterativen parameterbasierten Algorithmus zur Testfallableitung. *IPOG* generiert eine Testsuite basierend auf den Eingangsparametern und deren Diskretisierungsstufen mit einer *t-wise* (Übersetzung: t-weisen) Testabdeckung. Der Vorteil dieser Algorithmen besteht darin, dass neue Parameter und Diskretisierungsstufen zu einer bereits berechneten Testsuite hinzugefügt werden können, ohne die komplette Testsuite neu berechnen zu müssen. Ammann und Offutt (1994) haben den iterativen Algorithmus *base choice* vorgestellt, welcher basierend auf existierenden Testfällen die Testsuite erweitert. Dieser Algorithmus erstellt einen Basistest mit den wichtigsten Diskretisierungsstufen für jeden Parameter. Die wichtigste Diskretisierungsstufe kann beispielsweise der Wert sein, der am häufigsten auftritt oder bei dem eine hohe Fehlerwahrscheinlichkeit erwartet wird. Die Testsuite wird durch die Veränderung jeweils einer Diskretisierungsstufe eines Parameters um weitere Testfälle ergänzt.

#### 8.4.3.3 Empfehlung für einen kombinatorischen Algorithmus für den Test von automatisierten Fahrfunktionen

Basierend auf der Anzahl von Parametern und deren möglichen Diskretisierungsstufen, die einen Einfluss auf die automatisierte Fahrfunktionen haben, stellt der Test dieser Systeme eine große Herausforderung dar. Diese Systeme sollten aus Gründen der Sicherheit und Robustheit mit einer Testsuite getestet werden, welche eine *N-wise* (Übersetzung: N-weise) Testabdeckung hat. Dies ist allerdings aus ökonomischen und technischen Gründen nicht

umsetzbar (Sommerville, 2006, S.539). Daher ist ein Algorithmus für die kombinatorische Testfallableitung zu wählen, der eine paarweise oder *t-wise* (Übersetzung: t-weise) Testabdeckung erreicht, da nach Kuhn u. a. (2004) ein Fehler in vielen Fällen auf die Kombination von zwei Parameterwerten rückführbar ist. Da beim Test von automatisierten Fahrfunktionen die Szenarien und die Anzahl der Parameter sowie der Diskretisierungsstufen stark variieren, muss der Algorithmus mit unterschiedlichsten Testkonfigurationen Testfälle ableiten können. Daher qualifiziert sich nach aktuellem Forschungsstand der Algorithmus *IPOG* für die kombinatorische Testfallableitung. Dieser Algorithmus wird für die Ableitung der Testfälle im Fallbeispiel des Engstellenassistenten in Kapitel 12 verwendet.

## 8.5 Zusammenfassung

Im vergangenen Kapitel wurde die Testfallerstellung des modularen virtuellen Testbaukastens beschrieben, die einen zweistufigen Prozess darstellt. Im ersten Schritt werden Szenarien auf Basis der identifizierten Parameter beschrieben. Im zweiten Prozessschritt werden mittels der systematischen Testfallableitung aus logischen Szenarien konkrete Testfälle erstellt.

Zunächst wurden Anforderungen an die Beschreibung für Szenarien und die Testfallableitung für automatisierte Fahrfunktionen definiert. Daraufhin wurden ausgewählte Arbeiten aus dem Stand der Technik zur Szenariobeschreibung und zur Testfallableitung vorgestellt und anhand der gestellten Anforderungen bewertet. Dabei wurde festgestellt, dass strukturierte Modelle für die Beschreibung von Szenarien vorhanden sind. Jedoch fehlt in den Ansätzen die Möglichkeit einer Beschreibung für reproduzierbare Szenarien aus der Sicht des Testobjekts. Als möglicher Lösungsvorschlag wurde das hierarchische 4-Ebenen-Modell inklusive der Regelung der Akteure vorgestellt. Mit Hilfe des Modells wurde die Möglichkeit geschaffen, Szenarien in einer einheitlichen Struktur zu beschreiben. Das Modell wurde in den Entwurf des neuen Standards *OpenScenario* integriert. Durch die Regelung der Akteure mittels Manövern besteht die Möglichkeit, Szenarien mit einer lokalen, aus Sicht des Testobjekts, als auch globalen Reproduzierbarkeit zu beschreiben.

Mittels der systematischen Testfallableitung werden aus logischen Szenarien konkrete Testfälle abgeleitet. Die Testfallableitung nutzt ausschließlich Prüfmethode, die nicht den Quellcode des Testobjekts für die Ableitung benötigen. Durch die Anwendung der kombinatorischen Testfallableitung wird die Anzahl der redundanten Testfälle, wie sie in den Anforderungen beschrieben wurden, minimiert. Durch die vorgestellten Testabdeckungen besteht die Möglichkeit, je nach gewählter Abdeckung Fehler zu finden, die auf mehreren Diskretisierungsstufen basieren.

Wie in den Anforderungen dargestellt, stellt die Wahl der Testabdeckung für das jeweilige Testobjekt eine offene Forschungsfrage dar. Aufgrund der starken Reduktion der Testfälle bei der kombinatorischen Testableitung besteht die Möglichkeit, dass Wechselwirkungen von Diskretisierungsstufen, die über der kombinatorischen Testabdeckung liegen, nicht identifiziert werden. Daraus ergibt sich die Forschungsfrage nach der „richtigen“ kombinatorischen Testabdeckung für das jeweilige Testobjekt. Um einen Beitrag zur Beantwortung der Forschungsfrage zu leisten, wird die Anwendbarkeit der kombinatorischen Testfallableitung in Kapitel 12 anhand des Engstellenassistenten mit einer paarweisen Testabdeckung evaluiert. Anhand des Fallbeispiels wird analysiert, welche Effekte der Parameter sich aus der gewählten Testabdeckung ableiten lassen.





## 9 Testdurchführung mittels X-in-the-Loop-Verfahren

Dieses Kapitel beschreibt die Testdurchführung mit Hilfe des modularen virtuellen Testbaukastens. Für die Testdurchführung werden X-in-the-Loop-Verfahren verwendet. Zunächst werden die Anforderungen an die Testdurchführung definiert, um verschiedene Testdurchführungsmethoden zu bewerten. Darauf aufbauend wird ein relevanter Stand der Technik zur Testdurchführung beschrieben und mit den gestellten Anforderungen verglichen. Anschließend wird der Forschungsbedarf abgeleitet. Weiterhin wird eine Methode für die Zuordnung von Testfällen auf verschiedene X-in-the-Loop-Verfahren vorgestellt, die die Klassifikationsmethode aus Kapitel 4.3.2 verwendet. Neben der Zuordnungsmethode werden Testtreiber für Module der funktionalen Systemarchitektur aus Kapitel 3.2 vorgestellt. Dabei werden Herausforderungen erläutert und mögliche Lösungsansätze diskutiert.

### 9.1 Anforderungen

Das folgende Unterkapitel beschreibt die Anforderungen an die Testdurchführung und an die Verkehrssimulationen. Die Anforderungen wurden auf Basis der Erfahrungen des Autors im Projekt des Engstellenassistenten erhoben.

#### 9.1.1 Anforderungen an die Testdurchführung

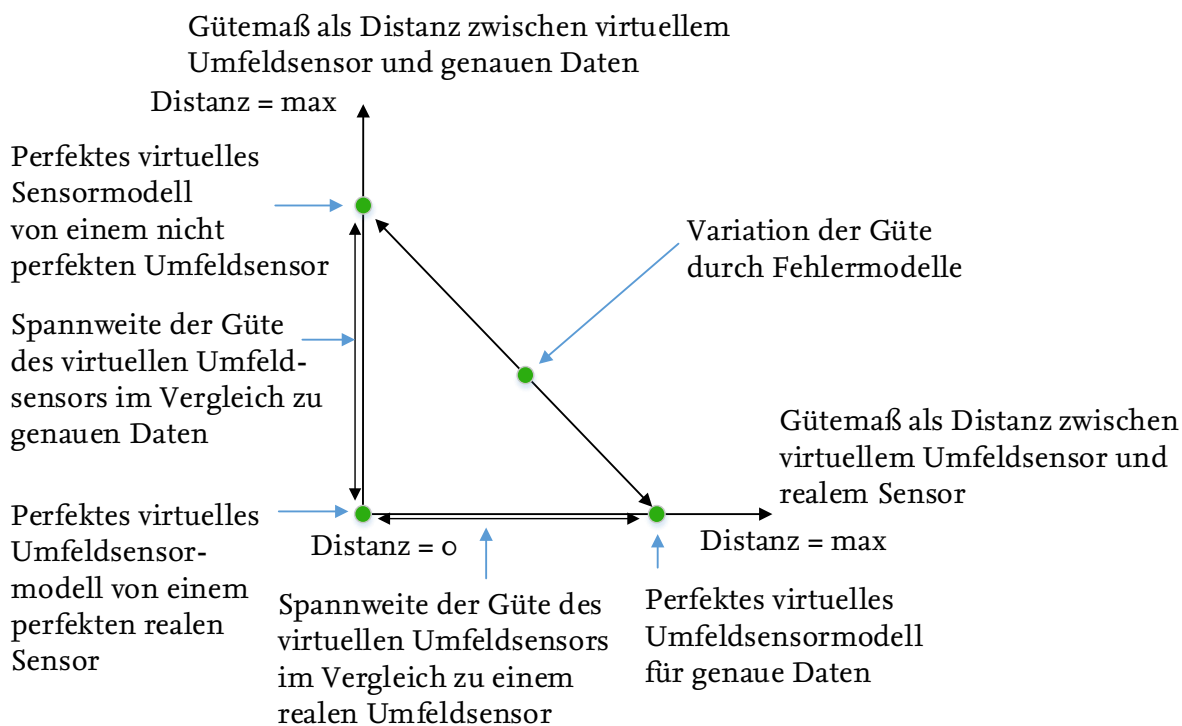
**Die Testdurchführung muss Eingangsdaten für das Testobjekt zur Verfügbarkeit stellen.**

Die Testdurchführung muss für die Teilmodule der automatisierten Fahrfunktion Eingangsdaten mit einer geforderten Frequenz zur Verfügung stellen. Zur Identifikation der verschiedenen Module und Schnittstellen sowie deren Eingangsdaten wird die in Kapitel 3.2 vorgestellte funktionale Systemarchitektur verwendet. Durch die Erfüllung dieser Anforderung wird die Möglichkeit geschaffen, die Funktionalität der Module der funktionalen Systemarchitektur zu testen. Weiterhin kann die Güte der Module bestimmt werden, indem die Ausgangsdaten des zu testenden Moduls mit den perfekten Eingangsdaten des nachfolgenden Moduls verglichen werden. Um die Güte der Module zu messen, werden Metriken benötigt, die von den jeweiligen Modulen abhängig sind und daher individuell gewählt werden müssen.

**Die Testdurchführung muss Information über die Güte der Eingangsdaten zur Verfügung stellen.**

Von der Testdurchführung wird gefordert, dass diese Eingangsdaten in verschiedenen Gütestufen zur Verfügung stellen kann. Dabei muss die Testdurchführung sowohl genaue Referenzdaten, die nach der DIN 5725 (2002) eine hohe Richtigkeit und Präzision haben, als auch Daten mit einer geringeren Güte zur Verfügung stellen. Die Information über die Güte der Eingangsdaten soll stets bekannt und verfügbar sein. Die Güte wird in der vorliegenden Arbeit in zwei Maßen unterschieden. Das erste Maß beschreibt die Distanz zwischen den Modelldaten und den genauen Referenzdaten. Das zweite Maß beschreibt die Distanz zwischen den Modelldaten und den Daten, die ein realer Umfeldsensor in der simulierten Szene

messen würde. Abbildung 9.1 verdeutlicht die beiden Gütemaße anhand eines Diagramms mit verschiedenen eingetragenen Beispielen. Über verschiedene Fehlermodelle sollen die Gütemaße variierbar sein. Dafür sollen die Fehlermodelle sowohl die Richtigkeit als auch die Präzision der Daten variieren. Die beiden Gütemaße sind direkt voneinander abhängig, wie in Abbildung 9.1 zu sehen. Eine Veränderung des ersten Gütemaßes führt direkt zu einer Änderung des zweiten Gütemaßes.



**Abbildung 9.1:** Gütemaße für die Simulationsmodelle

**Das Testobjekt darf bei der Entwicklung und Erprobung beziehungsweise Testdurchführung nicht gefährdet werden.**

Das Testobjekt darf bei der Erprobung nicht gefährdet oder beschädigt werden, um Wiederholungen der Tests zu ermöglichen. Dies ist besonders relevant beim Einsatz von kostenintensiven Prototypen oder beim Test von sicherheitsrelevanten Fahrfunktionen, die nach Funktionsdefinition erst auslösen oder eingreifen, um einen Unfall zu verhindern oder die Unfallschwere zu mindern, wie die automatische Notbremsung (ANB) (Reschka u. a., 2015b). Mit der Erfüllung dieser Anforderung wird ein Betrag zur Anforderung der Sicherheit an das Testkonzept geleistet.

**Die Testdurchführung muss reproduzierbare Testbedingungen herstellen.**

Die Testdurchführung muss in der Lage sein, Testfälle reproduzierbar durchzuführen, wobei hier auf die bereits in Kapitel 5.1.2 beschriebene Reproduzierbarkeit verwiesen wird. Durch die Erfüllung dieser Anforderung wird die Möglichkeit geschaffen, dass bei einer Wiederholung der Testfälle die gleichen Testergebnisse zu erwarten sind und somit die Fahrfunktion bei gleichen Eingaben das gleiche Verhalten zeigt.

**Die Testdurchführung muss mit und ohne Rückführung möglich sein.**

Von der Testdurchführung wird gefordert, dass Tests mit und ohne Rückführung der Ausgangsdaten des Testobjekts durchführbar sind. Die Testdurchführung ist in der Lage jeden Test ohne Rückführung auszuführen, wenn die Möglichkeit besteht, für jedes Modul die Eingangsdaten zur Verfügung zu stellen. Für die Durchführung von Tests mit Rückführung muss die Testdurchführung in der Lage sein, die Ausgangsdaten von jedem Modul zu verarbeiten, um daraus neue Eingangsdaten für das zu testende Modul oder die zu testende Verarbeitungskette zu erstellen. Somit müssen die übrigen Elemente der funktionalen Systemarchitektur, die nicht zum Testobjekt gehören, im Testtreiber durch vereinfachte Modelle repräsentiert werden.

**Die Testdurchführung soll parametrierbare Ausführungszeiten ermöglichen.**

Die Testdurchführung soll die Ausführung der Tests in verschiedenen Ausführungszeiten ermöglichen. Damit sollen die Tests schneller als in Echtzeit, exakt in Echtzeit als auch langsamer als in Echtzeit durchgeführt werden. Eine Durchführung der Tests schneller als in Echtzeit ist nur mit einer Testdurchführung möglich, die komplett in der Simulation ausgeführt wird. Dies gilt ebenfalls für Tests, die langsamer als in Echtzeit durchgeführt werden. Die Tests, die langsamer als in Echtzeit durchgeführt werden, bieten dem Anwender die Möglichkeit, Teile von Szenarien detaillierter zu untersuchen. Für eine Variation der Ausführungszeit muss das Testobjekt die Zeit der Durchführung variieren können. Dies setzt voraus, dass die einzelnen Verarbeitungsschritte des Testobjekts gezielt ausführbar sind und die Referenzzeit an der sich die Zeitstempel der Daten orientieren ebenfalls variierbar ist.

**Die Testdurchführung soll eine feingranulare Parametrisierung der Testparameter ermöglichen.**

Die Testdurchführung soll eine feingranulare Parametrisierung der Testparameter ermöglichen. Dies bedeutet, dass die vorgegebenen Parameterwerte beim Test eingehalten werden. Da die Parameterwerte je nach Testfall und Testobjekt stark variieren, sollen die Parameter in einer großen Spannweite des Parameterbereichs einstellbar sein. Mit der Erfüllung dieser Anforderung kann eine hohe Abdeckung des Testraums erreicht werden.

### 9.1.2 Anforderungen an Verkehrssimulationen

In Kapitel 4.2 wurden Verkehrssimulationen mit den Abstraktionsebenen *makroskopisch*, *mesoskopisch*, *mikroskopisch* und *submikroskopisch* vorgestellt. Die Tabelle 9.1 fasst die Eigenschaften der Ebenen anhand der Kriterien *Modellansatz*, *typische Beschreibungsgrößen*, *Schrittweiten der Berechnung*, *Anwendungsgebiete* und *Repräsentation der Straße* in Anlehnung an Weilkes (2000, S.60) zusammen. Als Erweiterung wurde die mesoskopische Ebene hinzugefügt.

Durch den Test der automatisierten Fahrfunktionen stellt der modulare virtuelle Testbaukasten Anforderungen an die Verkehrssimulation. Diese Anforderungen werden im Folgenden definiert, um darauf aufbauend eine Abstraktionsebene für den Einsatz im modularen virtuellen Testbaukasten auszuwählen.

Kriterium	Abstraktionsebene			
	makroskopisch	meso- skopisch	mikroskopisch	submikroskopisch
mögliche Modellansätze	Kontinuums- theorie	Zellulare Automaten	Fahrer- Fahrzeug- Einheit, Folgemodell des Fahrers	detaillierte Abbildung des Fahrers oder Fahrzeugs
typische Größen	Verkehrsstärke, Verkehrsdichte, mittlere Ge- schwindigkeit	mittlere Geschwin- digkeiten	Geschwindig- keits- und Wegverläufe, Zeitlücken	Geschwindigkeits- verläufe, Kraftstoffver- brauch
Schrittweite der Berechnungsschritte	10 s	1-10 s	1 s	<0,1 s
Anwendungsgebiete	Routen- umlegungen, Zuflussbe- schränkungen	Knoten- punkte	lokale Engpässe, Kolonnen- verhalten, Fahrer- beeinflussung	Antriebsstrang- auslegung, Verbrauchs- aussagen
Repräsentation des Umfelds	Graph mit Kanten- Knoten- Struktur	fahrstreifen- genaue, diskret- segmentierte Straßen	fahrstreifen- genaue, kontinuierliche Straße	fahrstreifengenaue, kontinuierliche Straße mit Oberflächenprofil

**Tabelle 9.1:** Abstraktionsebenen von Simulationen in Anlehnung an Weilkes (2000), erweitert um die mesoskopische Ebene

**Die Verkehrssimulation muss eine detaillierte Modellierung der Straßengeometrie und -topologie ermöglichen.**

Die Verkehrssimulation muss eine detaillierte Modellierung der Straßengeometrie und -topologie ermöglichen, um das 4-Ebenen-Modell in die Simulation zu übertragen. Dabei müssen unter anderem die Fahrstreifenbreiten und die Fahrstreifenmarkierungen quasi-kontinuierlich definierbar sein, um beispielsweise die in Kapitel 2.4 vorgestellten Regelquerschnitte von Straßen definieren zu können.

**Die Schrittweite der Berechnungsschritte der Verkehrssimulation muss unterhalb der Zykluszeit des Testobjekts sein.**

Die Schrittweite der Berechnungsschritte muss unterhalb der Zykluszeit des Testobjekts liegen, um die Möglichkeit zu schaffen, für jeden Zyklus des Testobjekts neue Eingangsdaten aus der Verkehrssimulationen bereitzustellen. In der vorliegenden Arbeit wird davon ausgegangen, dass die Zykluszeiten einer Fahrfunktion unterhalb von  $0,1 \text{ s}^1$  liegen.

<sup>1</sup>Dieser Wert beruht auf den Erfahrungen des Autors im Projekt des Engstellenassistenten.

**Die Verkehrssimulation muss kontinuierliche Bewegungen anderer Verkehrsteilnehmer darstellen.**

Die Bewegungen der übrigen Verkehrsteilnehmer müssen kontinuierlich in Längs- und Querrichtung darstellbar sein, um Bewegungen von Verkehrsteilnehmern innerhalb des eigenen Fahrstreifens zu simulieren. Eine Beispielanwendung wäre der Test von Ausweichmanövern, die aufgrund von anderen Verkehrsteilnehmern im eigenen Fahrstreifen notwendig sind.

**Die Verkehrssimulation muss eine getrennte Simulation von Fahrzeug, Fahrer und Fahrfunktion ermöglichen.**

Die Simulation von Fahrzeug, Fahrer und Fahrfunktion muss getrennt stattfinden. Dies ist beim Test von Fahrfunktionen mit den Automatisierungsgraden 1 - 3 relevant, um beispielsweise Tests von Übergabeszenarien zwischen Fahrfunktion und Fahrer zu ermöglichen. Daher muss für diese Tests eine getrennte Simulation von Fahrer und Fahrfunktion möglich sein.

**Auswahl einer Verkehrssimulation**

Makroskopische Verkehrssimulationen erfüllen keine der geforderten Anforderungen. Aufgrund des gewählten Modellansatzes der Kontinuumstheorie und der abstrakten Beschreibung der Straße ist keine detaillierte Modellierung der Straßengeometrie und -topologie möglich. Des Weiteren werden keine kontinuierlichen Bewegungen der Verkehrsteilnehmer dargestellt. Mesoskopische Verkehrssimulationen bieten die Möglichkeit der Modellierung von Straßengeometrien und -topologien. Durch die Darstellung der Straße in Form von diskreten Segmenten werden jedoch keine kontinuierlichen Bewegungen dargestellt. Des Weiteren liegt die Schrittweite von 10 s bis 1 s über der angenommenen Zykluszeit einer Fahrfunktion. Mikroskopische Verkehrssimulationen bieten die Möglichkeit der Darstellung von kontinuierlichen Bewegungen der übrigen Verkehrsteilnehmer. Allerdings genügt die Schrittweite der Berechnungsschritte von ca. 1 s nicht den Anforderungen des Testbaukastens. Des Weiteren ist keine getrennte Simulation von Fahrer, Fahrzeug und Fahrfunktion vorgesehen. Submikroskopische Verkehrssimulationen erfüllen die gestellten Anforderungen des Testbaukastens. Da eine getrennte Simulation von Fahrer, Fahrzeug und Fahrfunktion stattfindet, besteht die Möglichkeit, Übergabeszenarien zwischen Fahrer und Fahrfunktion zu testen. Durch die detaillierte Modellierung der Straßengeometrie und -topologie kann die Beschreibung des Szenarios im 4-Ebenen-Modell in die Simulation übertragen werden. Aus diesem Grund wird für den Test von automatisierten Fahrfunktionen im modularen virtuellen Testbaukasten eine submikroskopische Simulation benötigt. Die Tabelle 9.2 fasst die gestellten Anforderungen und die Erfüllung der Anforderungen durch die verschiedenen Verkehrssimulationen zusammen.



Anforderung	Abstraktionsebene			
	makro- skopisch	meso- skopisch	mikro- skopisch	submikro- skopisch
Modellierung der Straengeometrie und -topologie	✗	✓	✓	✓
Schrittweite der Berechnungsschritte unterhalb der Zykluszeit des Testobjekts	✗	✗	✗	✓
Kontinuierliche Bewegungen anderer Verkehrsteilnehmer	✗	✗	✓	✓
Getrennte Simulation von Fahrer, Fahrzeug und Fahrfunktion	✗	✗	✗	✓

**Tabelle 9.2:** Gestellte Anforderungen an die Abstraktionsebenen der Verkehrssimulation durch den modularen virtuellen Testbaukasten. Legende: ✓: erfllte Anforderung, ✗: nicht erfllte Anforderung.

## 9.2 Stand der Technik

Dieses Unterkapitel betrachtet den Stand der Technik zur Testdurchfhrung fr automatisierte Fahrfunktionen. Fr die Testdurchfhrung besteht die Mglichkeit, die Tests in der Simulation, auf einem abgesperrten Testgelnde oder im realen Straenverkehr durchzufhren. Die Methoden der Testdurchfhrung werden anhand der Erfllung der gestellten Anforderungen bewertet. Aus der Bewertung wird der Forschungsbedarf abgeleitet.

### 9.2.1 Realtests im ffentlichen Straenverkehr

Bei den Realtests im realen Straenverkehr wird die automatisierte Fahrfunktion durch eine fehlerfreie Funktionalitt whrend einer festgelegten Anzahl von Testkilometern abgesichert. Durch die Anzahl der gefahrenen Testkilometer kann abgeschtzt werden, wie gut das System im Verhltnis zu menschlichen Fahrern ist (Wachenfeld und Winner, 2015a).

Die Tests im ffentlichen Straenverkehr werden in Tests mit einem einzelnen Versuchsfahrzeug und in Feldtests mit einer Vielzahl von Versuchstrgern, die sich gleichzeitig im ffentlichen Straenverkehr bewegen, unterteilt. Feldtests haben den Vorteil, dass das System gleichzeitig unter verschiedensten Randbedingungen, wie unterschiedlichen Fahrern oder Verkehrssituationen, getestet wird. Feldtests stellen eine Parallelisierung der Testdurchfhrung dar, wodurch in gleicher Zeit mehr Testflle durchgefhrt werden. Jedoch steigen durch die gleichzeitig bentigten Versuchsfahrzeuge die Kosten fr die Durchfhrung. Das EU-Projekt *euroFOT*<sup>2</sup> stellt ein Beispiel fr einen ffentlich gefrderten Feldtest dar. Benmimoun u. a. (2012b) beschreiben das Projekt und zeigen die Nutzung der Daten fr eine Auswertung der Effekte von ACC und Kollisionswarnungen auf die Geschwindigkeit und den Kraftstoffverbrauch der Versuchsfahrzeuge.

Beispiele fr Tests im ffentlichen Straenverkehr haben unter anderem die Projekte *Stadtpilot* (Saust u. a., 2009, 2011), *Google Self-Driving Car Project* (Korosec, 2016), *Audi Piloted Driving: Jack*<sup>3</sup> (Ulbrich u. a., 2016) und *BRAiVE* (Broggi u. a., 2013) gezeigt. Aktuelle

<sup>2</sup><http://www.eurofot-ip.eu/>, abgerufen: 04.03.2016

<sup>3</sup>[http://www.audi.com/com/brand/en/vorsprung\\_durch\\_technik/content/2014/10/piloted-driving.html](http://www.audi.com/com/brand/en/vorsprung_durch_technik/content/2014/10/piloted-driving.html), abgerufen: 30.04.2016

Bestrebungen von offiziellen Behörden zeigen, wie relevant die Realtests im öffentlichen Straßenverkehr sind. Mit dem Projekt *Digitales Testfeld Autobahn*<sup>4</sup> wird die Autobahn A9 zu einer öffentlichen Teststrecke ausgebaut. In Wuppertal werden Teile der Landesstraße L418 für den Test von automatisierten Fahrfunktionen freigegeben (Eisert, 2015). Die SAE (2015) stellt Richtlinien für die Durchführung von Realtests im öffentlichen Straßenverkehr vor. Es werden beispielsweise Hinweise zur Ausbildung der Testfahrer und zur Auswahl von Testrouten gegeben. Die Kraftfahrzeugbehörde von Kalifornien (engl. Department of Motor Vehicles) (DMV, 2014) schreibt Richtlinien für den Test von automatisierten Fahrfunktionen in Kalifornien vor. Eine Richtlinie ist beispielsweise die Erstellung von öffentlichen Berichten über Unfälle, die sich ereignen, während die Fahrfunktion aktiv ist. So wurde ein Unfallbericht eines Fahrzeugs vom Projekt *Google Self-Driving Car* von der Kraftfahrzeugbehörde (DMV, 2016) veröffentlicht.

### **Bewertung der gestellten Anforderungen**

**Verfügbarkeit von Eingangsdaten und deren Güte:** Realtests haben den Vorteil, dass keine Systembestandteile simuliert werden und somit keine vereinfachten Modelle verwendet werden, um die Eingangsdaten für die einzelnen Systembestandteile zu erstellen. Daher besitzen die Eingangsdaten für die Verarbeitungskette des Testobjekts die maximale Güte bezüglich der Realitätsnähe.

**Gefährdung während der Tests:** Da die Realtests im öffentlichen Straßenverkehr durchgeführt werden, besteht eine bedingte Gefahr für das Versuchsfahrzeug, die Personen im Versuchsfahrzeug als auch für die übrigen Verkehrsteilnehmer um das Versuchsfahrzeug. Ein gegebenenfalls nicht vorhersehbares oder kontrollierbares Verhalten der Fahrfunktion kann zu einer Gefährdung des Versuchsfahrzeugs oder der anderen Verkehrsteilnehmer führen, wenn ein vorhandener Sicherheitsfahrer nicht rechtzeitig eingreifen kann. Des Weiteren kann das Verhalten der übrigen Verkehrsteilnehmer nur bedingt abgeschätzt werden, sodass von diesen ebenfalls eine Gefahr ausgehen kann.

**Reproduzierbare Testbedingungen:** Weiterhin haben Realtests im öffentlichen Straßenverkehr den Nachteil, dass die Testdurchführung durch das individuelle Verhalten der übrigen Verkehrsteilnehmer zum Teil stochastisch ist und daher nur bedingt reproduzierbare Testbedingungen bietet. Je nachdem wie sich die umliegenden Verkehrsteilnehmer verhalten, ergeben sich unterschiedlichste Szenarien für die Fahrfunktion. Für allgemein gültige Verkehrsregeln, wie die Straßenverkehrsordnung, kann jedoch davon ausgegangen werden, dass diese von den übrigen Verkehrsteilnehmern in der Regel reproduzierbar eingehalten werden. Dadurch kann bei dieser Art der Testdurchführung nur zum Teil von Zufallstests gesprochen werden.

**Tests mit und ohne Rückführung:** Realtests im öffentlichen Straßenverkehr können ohne Rückführung der Ausgänge des Testobjekts, wie von Winner (2002) vorgestellt, durchgeführt werden. Bei diesen Tests erhält das Testobjekt während der Fahrt alle benötigten Eingangsdaten, jedoch werden die Ausgangsdaten nicht an die nachfolgenden Module weitergegeben. Bei Tests mit Rückführung handelt es sich um Tests, wie sie üblicherweise durchgeführten werden, um das Gesamtsystem zu testen. Dabei werden die Ausgangsdaten an die nachfolgenden Module zur Verarbeitung weitergegeben.

---

<sup>4</sup><http://www.bmvi.de/digitalestestfeldautobahn>, abgerufen: 04.03.2016

**Einstellbare Ausführungszeiten:** Realtests im öffentlichen Straßenverkehr sind nur in Echtzeit möglich, wodurch die Ausführungszeit weder erhöht noch reduzierbar ist. Die einzige Möglichkeit, die Ausführungszeit aller geplanten Testfälle zu verringern, ist die Parallelisierung der Ausführung durch die Erhöhung der Anzahl der eingesetzten Versuchsfahrzeuge (Cao u. a., 2015). Damit wird jedoch die Ausführungszeit für einen einzelnen Testfall nicht verändert.

**Feingranulare Parametrisierung der Testparameter:** Realtests bieten nur eine bedingte Möglichkeit der feingranularen Parametrisierung der Testfälle, da die Testparameter von der gewählten Umgebung (öffentlicher Straßenverkehr) und deren Bedingungen, wie Tageszeit, Wetter, Straßenverhältnisse, und den übrigen Verkehrsteilnehmern abhängig sind. Dies stellt allerdings gleichzeitig einen Vorteil dar, da durch die zufällige Variation der Eingangsparameter Testfälle durchgeführt werden, die eventuell bei einer systematischen Variation der Parameter nicht identifiziert worden wären.

Tabelle 9.3 fasst die Erfüllung der Anforderungen an die Testdurchführung durch Realtests im öffentlichen Straßenverkehr zusammen.

Anforderung	Erfüllung
Verfügbarkeit von Eingangsdaten	✓
Information über die Güte der Daten	✓
Keine Gefährdung für das Testobjekt	○
Reproduzierbare Testbedingungen	○
Tests mit und ohne Rückführung	✓
Einstellbare Ausführungszeiten	✗
Feingranulare Parametrisierung der Testparameter	○

**Tabelle 9.3:** Übersicht der Erfüllung der Anforderungen der Testdurchführung bei Realtests auf öffentlichen Straßen. Legende: ✓: erfüllte Anforderung, ○: bedingt erfüllte Anforderung, ✗: nicht erfüllte Anforderung.

### 9.2.2 Realtests auf dem Testgelände

Eine Alternative zum Realtest im öffentlichen Straßenverkehr stellen die Testdurchführungen mit einem realen Versuchsträger auf dem Testgelände dar. Dabei wird Testequipment, wie Ballon-Fahrzeuge oder andere Elemente, wie der von Fecher u. a. (2015) vorgestellte Zielsimulator für Antikollisionssysteme (*EVITA*), zur Emulation der Umgebung verwendet. Bei der Planung und Erstellung der ersten Testgelände der Automobilhersteller lag der Fokus vor allem auf dem Test der fahrdynamischen Eigenschaften und der Mechanik des Gesamtfahrzeugs (Bagschik u. a., 2015b).

Neuere Testgelände haben den Fokus auf der funktionalen Absicherung beziehungsweise dem funktionalen Test von automatisierten Fahrfunktionen. Hierfür werden Szenarien getestet, die auf öffentlichen Straßen nicht ohne Gefährdung von anderen Verkehrsteilnehmern beziehungsweise nicht reproduzierbar durchführbar sind. Für die Testdurchführung werden unter anderem Nachbildungen von städtischen Szenarien mit verschiedenen Straßenmarkierungen und beweglichen Elementen, wie Fußgänger- beziehungsweise Tiernachbildungen

oder ferngesteuerten Fahrzeugen, bei denen Kollisionen zu keinen Beschädigungen führen, aufgebaut. Des Weiteren werden Elemente der Szenerie verwendet, bei denen bekannt ist, dass verschiedene Arten von Umfeldsensoren erhöhte *falsch-positiv* oder *falsch-negativ* Detektionen haben, wie Tunnel, Unterführungen oder Brücken. Dadurch werden Umfeldsensoren oder die vollständige Fahrfunktion gezielt in Szenarien getestet, die im öffentlichen Straßenverkehr Herausforderungen darstellen. Beispiele für solche Testgelände sind das von der Universität Michigan betriebene Testgelände *MCity*<sup>5</sup>, auf dem der Automobilhersteller Ford seine automatisiert fahrenden Fahrzeuge testet<sup>6</sup>, das Testgelände *Castle Air Force Base*, welches von Google<sup>7</sup> genutzt wird, das Testgelände der Firma Autoliv<sup>8</sup> oder das Testgelände *GoMentum Station*<sup>9</sup>, bei dem unter anderem die Firma Apple nach einem Zugang gefragt hat<sup>10</sup>.

### Bewertung der gestellten Anforderungen

**Verfügbarkeit von Eingangsdaten und deren Güte:** Die Verfügbarkeit der Eingangsdaten ist vollständig gegeben, da die komplette Fahrfunktion für die Durchführung der Tests zur Verfügung stehen muss. Die Güte der Eingangsdaten ist gegebenenfalls geringer als bei der Testdurchführung im öffentlichen Straßenverkehr, da die Tests in einer nachgebauten Umgebung durchgeführt werden. Je nachdem, welche Emulation genutzt wird, kann es beispielsweise bei Umfeldsensoren zu einem veränderten Detektionsverhalten kommen. Dies ist bei der Testdurchführung auf dem Testgelände zu beachten.

**Gefährdung während der Tests:** Da die Tests in einer geschützten Umgebung durchgeführt werden, ist das Testobjekt im Vergleich zur Testdurchführung im öffentlichen Straßenverkehr einer geringeren Gefährdung ausgesetzt. Jedoch interagiert das Versuchsfahrzeug während der Versuche mit der Realität, sodass von einer Restgefährdung für das Testobjekt ausgegangen wird. Dies gilt insbesondere bei Tests bei denen auf die Sicherheitsfahrer verzichtet wird, da der Test für diese zu gefährlich ist. In diesem Fall kann bei einem Fehlverhalten nur bedingt eingegriffen werden. Bei der Wahl der Emulation der Szene sollte ein Kompromiss zwischen der Gefährdung des Testobjekts und der Güte der Eingangsdaten getroffen werden.

**Reproduzierbare Testbedingungen:** Bei der Testdurchführung auf dem Testgelände wird gegenüber der Testdurchführung im öffentlichen Straßenverkehr von einer höheren Reproduzierbarkeit ausgegangen, da die Möglichkeit besteht, auf mehr Faktoren Einfluss zu nehmen. Der Start des Szenarios sowie der genaue Ablauf können gezielt gesteuert werden. Jedoch sind zahlreiche Faktoren der Tests nur bedingt beeinflussbar. Dazu zählen unter anderem das Wetter, der Wind und die Temperatur der Umgebung, die zum Beispiel nur indirekt durch die Wahl des Testtages beeinflusst werden können. Dadurch sind die Szenarien auf dem Testgelände nur bedingt reproduzierbar.

<sup>5</sup><http://college.usatoday.com/2015/07/20/university-of-michigan-opens-test-city-for-autonomous-cars/>, abgerufen: 09.12.2015

<sup>6</sup><http://www.usatoday.com/story/tech/2015/11/13/ford-revs-up-self-driving-car-program-fake-michigan-city/75673552>, abgerufen: 09.12.2015

<sup>7</sup><https://backchannel.com/license-to-not-drive-6dbea84b9c45>, abgerufen: 04.03.2016

<sup>8</sup><http://www.testcenter.autoliv.com/EN/ActiveSafety/Pages/default.aspx>, abgerufen: 07.03.2016

<sup>9</sup><http://gomentumstation.net/vision>, abgerufen: 19.04.2016

<sup>10</sup><https://www.theguardian.com/technology/2015/aug/14/apple-self-driving-car-project-titan-sooner-than-expected>, abgerufen: 19.04.2016

**Tests mit und ohne Rückführung:** Die Tests auf dem Testgelände können, wie bei der Testdurchführung im öffentlichen Straßenverkehr, mit und ohne Rückführung durchgeführt werden.

**Einstellbare Ausführungszeit:** Die Tests können, wie bei der Testdurchführung im öffentlichen Straßenverkehr, nur in Echtzeit durchgeführt werden.

**Feingranulare Parametrisierung der Testparameter:** Wie schon bei der Reproduzierbarkeit der Testbedingungen beschrieben, kann bei der Testdurchführung auf dem Testgelände auf die Testparameter zum Teil Einfluss genommen werden. Dadurch ist es möglich, eine Parametrisierung der Testfälle vorzunehmen. Der mögliche Parameterbereich und deren Parametrisierung hängt vom eingesetzten Testequipment ab. Je nach Equipment können beispielsweise verschiedene Relativgeschwindigkeiten oder Abstände zwischen verschiedenen Objekten während der Testdurchführung eingestellt werden.

Tabelle 9.4 fasst die Erfüllung der Anforderungen an die Testdurchführung durch Realtests auf dem Testgelände zusammen.

Anforderung	Erfüllung
Verfügbarkeit von Eingangsdaten	✓
Information über die Güte der Daten	○
Keine Gefährdung für das Testobjekt	○
Reproduzierbare Testbedingungen	○
Tests mit und ohne Rückführung	✓
Einstellbare Ausführungszeiten	✗
Feingranulare Parametrisierung der Testparameter	○

**Tabelle 9.4:** Übersicht der Erfüllung der Anforderungen der Testdurchführung bei Realtests auf dem Testgelände. Legende: ✓: erfüllte Anforderung, ○: bedingt erfüllte Anforderung, ✗: nicht erfüllte Anforderung.

### 9.2.3 X-in-the-Loop-Verfahren

Nachdem in den vorherigen Unterkapiteln die Testdurchführung im öffentlichen Straßenverkehr und auf dem Testgelände bewertet wurden, werden im Folgenden die in Kapitel 4.3.3 vorgestellten X-in-the-Loop-Verfahren bewertet. Einige X-in-the-Loop-Verfahren benötigen für die Testdurchführung eine reale Teststrecke und könnten somit den Testgeländetests zugeordnet werden. Jedoch nehmen die Verfahren im modularen virtuellen Testbaukasten durch das Modul der Testdurchführung eine besondere Bedeutung ein, sodass die Verfahren in diesem Kapitel besonders betrachtet und bewertet werden.

Veröffentlichungen zeigen, dass verschiedene X-in-the-Loop-Verfahren bereits für die Testdurchführung von verschiedenen Fahrfunktionen eingesetzt werden. Nentwig und Stamminger (2010, 2011) stellen beispielsweise den Test eines Fahrstreifenhalteassistenten in der Simulation vor. Laschinsky u. a. (2010) zeigen den Test eines aktiven Lichtsystems und Berger u. a. (2014a,b) stellen den Test von EuroNCAP<sup>11</sup> Szenarien in der Simulation vor. Die X-in-

<sup>11</sup>EuroNCAP ist eine Abkürzung für *European New Car Assessment Programme*. Frei übersetzt: Europäisches Neuwagen-Bewertungs-Programm



the-Loop-Verfahren können nach Strasser (2012, S.24) auf verschiedenen Prüfebenen des V-Modells eingesetzt werden und haben je nach Verfahren individuelle Vor- und Nachteile. Um diese Vor- und Nachteile zu identifizieren, werden im Folgenden die in Kapitel 4.3.3 klassifizierten Testwerkzeuge anhand der gestellten Anforderungen bewertet.

### **Bewertung der gestellten Anforderungen**

**Verfügbarkeit von Eingangsdaten und deren Güte:** Die Verfügbarkeit von Eingangsdaten ist bei allen X-in-the-Loop-Verfahren direkt von den verfügbaren Simulationsmodellen abhängig. Da nicht sichergestellt werden kann, dass die jeweiligen Modelle zur Erstellung der Eingangsdaten zur Verfügung stehen, wird diese Anforderungen nur bedingt erfüllt. Weiterhin fehlen zurzeit Metriken, um die Güte der Simulationsmodelle und deren Ausgangsdaten zu bewerten. Bei verfügbaren Simulationsmodellen sollen Eingangsdaten für das Testobjekt in verschiedenen Gütestufen generiert werden. Jedoch ist nur bei genauen Daten die Güte der Daten bekannt. Bei einer Reduzierung der Güte beispielsweise durch Fehlermodelle kann aufgrund der fehlenden Metriken keine Aussage über die Güte getroffen werden. Daher wird die Anforderung an die Information über die Güte der Daten bedingt erfüllt.

**Gefährdung während der Tests:** Eine Gefährdung für das Testobjekt kann nur ausgeschlossen werden, wenn bei der Testdurchführung keine realen oder emulierten dynamischen Elemente im Testaufbau vorhanden sind. Dies trifft für die X-in-the-Loop-Verfahren *Software-in-the-Loop*, *Driver-in-the-Loop* und *Hardware-in-the-Loop* zu. Bei den übrigen Testwerkzeugen besteht die Möglichkeit, dass sich dynamische Elemente um das Testobjekt bewegen oder sich das Testobjekt selbst bewegt, sodass eine Gefährdung für das Testobjekt nicht ausgeschlossen ist. Da die Versuche jedoch zum Teil auf einem Testgelände durchgeführt werden, kann eine Gefährdung durch das unerwartete Verhalten fremder Verkehrsteilnehmer ausgeschlossen werden. Daher wird die Anforderung bedingt erfüllt.

**Reproduzierbare Testbedingungen:** Reproduzierbare Testbedingungen für das Testobjekt werden erreicht, wenn alle Dimensionen der Klassifikation bis auf das Testobjekt simuliert werden. Wenn reale oder emulierte Elemente im Test vorhanden sind, wird nur eine bedingte Reproduzierbarkeit erreicht, da sich real vorhandene oder emulierte Elemente gegebenenfalls bei den Testdurchführungen unterschiedlich verhalten. Beispielsweise wird durch den Einsatz eines Fahrroboters beim *koordinierten automatisierten Fahren* (Schöner u. a., 2011) die Reproduzierbarkeit der Testbedingungen erhöht. Jedoch ist das Fahrzeug real vorhanden, wodurch die Reproduzierbarkeit nicht sichergestellt werden kann. Dadurch kann nur mit den Verfahren *Software-in-the-Loop* und *Hardware-in-the-Loop* die Anforderung der reproduzierbaren Testbedingungen erfüllt werden. Bei den übrigen Verfahren werden durch eine Emulation der eingreifenden Elemente, wie ein Fahrroboter, oder eine Regelung von dynamischen Elementen, wie in Kapitel 8.3.3 vorgestellt, bedingt reproduzierbare Testbedingungen erreicht.

**Tests mit und ohne Rückführung:** Die Erfüllung dieser Anforderung hängt von der Verfügbarkeit der Eingangsdaten ab, da für die Testdurchführungen die Eingangsdaten benötigt werden. Diese Anforderung wird bei allen X-in-the-Loop-Verfahren bedingt erfüllt, da bei allen Verfahren die Anforderung nach der Verfügbarkeit der Eingangsdaten bedingt erfüllt wird. Für Tests mit einer Rückführung der Daten des Testobjekts werden Simulationsmodelle benötigt, die die Ausgangsdaten des Testobjekts zu neuen Eingangsdaten für das Testobjekt verarbeiten. Bei den Verfahren *Vehicle-Hardware-in-the-Loop*, *Vehicle-*



*in-the-Loop*, *MOBILE*, *InDrive* und *koordinierten automatisierten Fahren* können Tests mit einer Rückführung durchgeführt werden, da bei diesen die komplette Fahrfunktion zur Verfügung stehen muss und somit keine verarbeitenden Modelle benötigt werden. Bei den übrigen Verfahren ist die Erfüllung der Anforderung verknüpft mit der Verfügbarkeit der verarbeitenden Simulationsmodelle. Daher wird die Anforderung bei diesen Verfahren bedingt erfüllt.

**Einstellbare Ausführungszeit:** Die Ausführungszeit der Tests ist lediglich beim Verfahren *Software-in-the-Loop* veränderbar, da bei diesem Verfahren alle Dimensionen simuliert werden. Jedoch ist die Ausführungszeit nur bedingt veränderbar, da diese von der Ausführungszeit des Testobjekts und den verwendeten Simulationsmodellen abhängig ist. Sollten die Berechnungen des Testobjekts schneller oder langsamer als in Echtzeit durchführbar sein, ist die Ausführungszeit variierbar. Anderenfalls besteht lediglich die Möglichkeit, die Testfälle in Echtzeit durchzuführen. Bei den übrigen Verfahren ist eine Durchführung der Tests nur in Echtzeit möglich, da immer mindestens eine Dimension emuliert wird oder real vorhanden ist.

**Feingranulare Parametrisierung der Testparameter:** Eine feingranulare Parametrisierung der Einflussparameter ist bei Verfahren möglich, bei denen alle Dimensionen bis auf das Testobjekt simuliert werden, da in diesem Fall die Parameter in einem weiten feingranularen Wertebereich einstellbar sind. Dies trifft auf die Verfahren *Software-in-the-Loop* und *Hardware-in-the-Loop* zu. Bei den übrigen Verfahren ist mindestens eine Dimension emuliert oder real vorhanden. Dadurch sind die Einflussparameter in den Grenzen und Diskretisierungsstufen der Spezifikation des Testwerkzeugs parametrierbar. Daher wird bei den übrigen Verfahren die Anforderung bedingt erfüllt.

Tabelle 9.5 fasst die Erfüllung der Anforderungen an die Testdurchführung durch die klassifizierten X-in-the-Loop-Verfahren zusammen.

Anforderung	Erfüllung							
	SIL	HIL	DIL	VEHIL	VIL	MOBILE	InDrive	KaF
Verfügbarkeit von Eingangsdaten	○	○	○	○	○	○	○	○
Information über die Güte der Daten	○	○	○	○	○	○	○	○
Keine Gefährdung für das Testobjekt	✓	✓	✓	○	○	○	○	○
Reproduzierbare Testbedingungen	✓	✓	○	○	○	○	○	○
Tests mit und ohne Rückführung	○	○	○	✓	✓	✓	✓	✓
Einstellbare Ausführungszeiten	○	✗	✗	✗	✗	✗	✗	✗
Feingranulare Parametrisierung der Testparameter	✓	✓	○	○	○	○	○	○

**Tabelle 9.5:** Übersicht der Erfüllung der Anforderungen der Testdurchführung bei Tests in der Simulation. Legende: ✓: erfüllte Anforderung, ○: bedingt erfüllte Anforderung, ✗: nicht erfüllte Anforderung. Abkürzungen: SIL: Software-in-the-Loop, HIL: Hardware-in-the-Loop, DIL: Driver-in-the-Loop, VEHIL: Vehicle-Hardware-in-the-Loop, VIL: Vehicle-in-the-Loop, KaF: Koordiniertes automatisiertes Fahren

### 9.2.4 Bewertung der Ansätze und Ableitung des Forschungsbedarfs

Die Bewertung zeigt, dass jedes Verfahren der Testdurchführung seine individuellen Vor- und Nachteile beziehungsweise Stärken und Schwächen besitzt. Die Realtests bestechen durch ihre genaue Güte der Eingangsdaten aufgrund der nicht benötigten Simulationsmodelle für die Erzeugung der Eingangsdaten. Jedoch ist die Durchführung der Tests mit hohen Kosten und Zeitaufwand verbunden, da die Durchführung der Tests nur in Echtzeit möglich ist. Eine Reduzierung der Ausführungszeit aller Testfälle kann durch eine Parallelisierung der Testdurchführung mittels mehrerer Versuchsträger erreicht werden. Weiterhin ist eine feingranulare Parametrisierung der Testparameter nur bedingt möglich und die Versuche nur zum Teil reproduzierbar.

Die X-in-the-Loop-Verfahren haben den Nachteil, dass derzeit Metriken fehlen, um die Güte der Ausgangsdaten der verwendeten Simulationsmodelle zu messen. Aufgrund dieser fehlenden Metriken sind auch in Zukunft Tests im öffentlichen Straßenverkehr und auf dem Testgelände notwendig. Die Vorteile der X-in-the-Loop-Verfahren sind die bedingte Reproduzierbarkeit, die feingranulare Parametrisierung der Testparameter sowie die bedingte Gefährdung des Testgegenstands bei der Durchführung der Tests. Bei der Bewertung der Verfahren wurde festgestellt, dass lediglich beim *Software-in-the-Loop*-Verfahren eine Testdurchführung schneller oder langsamer als in Echtzeit möglich ist. Dafür müssen die Algorithmen schneller als in Echtzeit durchführbar sein. Anderenfalls ist auch beim *Software-in-the-Loop*-Verfahren nur eine Testdurchführung in Echtzeit möglich.

Die Testdurchführung soll eine effektive und effiziente Durchführung der Tests ermöglichen. Dies bedeutet nach den Definitionen aus Kapitel 2.1.2 und 2.1.3, dass die Testdurchführung den Anwender bei der Auswahl eines Testwerkzeugs unterstützt und ein Testwerkzeug für den jeweiligen Testfall empfiehlt. Dabei soll eine Reduktion der notwendigen Kosten und der Zeit für die Ausführung der Testfälle erreicht werden. Weiterhin soll das ausgewählte Werkzeug eine geforderte Ergebnisgüte erreichen, um die Ergebnisse der Tests interpretieren zu können. Um dieses Ziel zu erreichen, ist eine Methode zu entwickeln, die eine effiziente und effektive Zuordnung der Testfälle auf die verfügbaren Testwerkzeuge ermöglicht. Ein erster Ansatz einer solchen Zuordnungsmethode wird im folgenden Kapitel vorgestellt. Die Grenze der Betrachtung für die Zuordnungsmethode wird um die klassifizierten X-in-the-Loop-Verfahren gezogen. Testdurchführungen auf dem Testgelände und im öffentlichen Straßenverkehr werden nicht weiter untersucht, da der Testbaukasten ausschließlich die Testdurchführung in Kombination mit Simulation und Emulationen betrachtet.

Bei der Bewertung der X-in-the-Loop-Verfahren wurde weiterhin aufgezeigt, dass die Eingangsdaten für das Testobjekt nur bedingt verfügbar sind. Die Verfügbarkeit der Eingangsdaten hängt von der Verfügbarkeit der Simulationsmodelle ab, die für die Erstellung der Eingangsdaten benötigt werden. Daher fehlt eine Bewertung, mit welchen Modellen die Eingangsdaten für die jeweiligen Testobjekte erstellbar sind, um so die Einsatzmöglichkeiten der Simulation beim Test von verschiedenen Testobjekten zu bewerten. Diese Bewertung wird in Kapitel 9.4 anhand der Module der funktionalen Systemarchitektur aus Kapitel 3.2 vorgestellt. Dazu werden benötigte Testtreiber identifiziert, die für den Test mit und ohne Rückführung benötigt werden. Weiterhin werden die resultierenden Herausforderungen für die Testdurchführung diskutiert.

## 9.3 Eine Methode für die Zuordnung von Testfällen auf X-in-the-Loop-Verfahren

Dieses Unterkapitel beschreibt eine Methode für die Zuordnung von Testfällen auf X-in-the-Loop-Verfahren. Die Zuordnungsmethode wird anhand der Fahrfunktion aus Kapitel 3.3 exemplarisch angewendet.

Teile dieses Kapitels wurden bereits in Schuldt u. a. (2015b)<sup>12</sup> veröffentlicht.

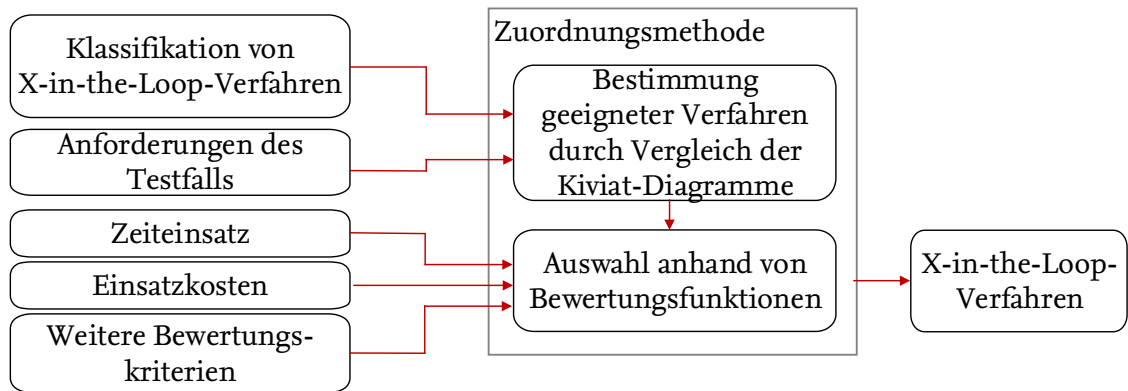
### 9.3.1 Beschreibung der Zuordnungsmethode

Die Zuordnungsmethode stellt ein zweistufiges Verfahren dar. Auf der ersten Stufe werden geeignete X-in-the-Loop-Verfahren für den Testfall klassifiziert. Basierend auf dieser Vorselektion wird im zweiten Schritt das optimale Verfahren anhand einer Bewertungsfunktion ausgewählt. Abbildung 9.2 zeigt den schematischen Ablauf der Zuordnungsmethode.

Für die Anwendung der Zuordnungsmethode werden drei Arten von Eingangsdaten benötigt. Zunächst werden die in Kapitel 4.3.2 vorgestellten Klassifikationen der X-in-the-Loop-Verfahren benötigt. Weiterhin werden die Anforderungen des Testfalls an die X-in-the-Loop-Verfahren betrachtet. Diese Anforderungen definieren die Simulation, Emulation oder das Realvorhandensein von Elementen im Testfall. Die Anforderungen werden anhand der identifizierten Dimensionen zur Klassifikation der X-in-the-Loop-Verfahren in Form

---

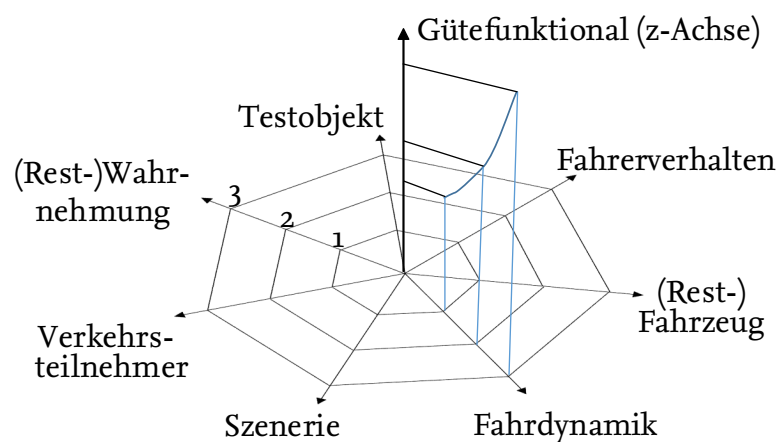
<sup>12</sup>Die Idee der Zuordnungsmethode durch die Klassifikation mittels Kiviat-Diagramme wurden vom Autor der vorliegenden Arbeit erbracht. Die Koautoren haben durch intensive Diskussionen und Reviews zur Reife und Detaillierung des Ansatzes beigetragen.



**Abbildung 9.2:** Zuordnungsmethode für Testfälle auf X-in-the-Loop-Verfahren

eines Kiviat-Diagramms beschrieben. Die dritten Eingangsdaten beschreiben die Güte der verfügbaren Modelle, den Zeiteinsatz und die Kosten der Durchführung und weitere Bewertungskriterien der X-in-the-Loop-Verfahren für die jeweiligen Diskretisierungsstufen.

Mittels der ersten beiden Eingangsdaten werden geeignete und ungeeignete X-in-the-Loop-Verfahren für die Durchführung des Testfalls identifiziert. Ein X-in-the-Loop-Verfahren ist für die Durchführung des Testfalls geeignet, wenn sich das Kiviat-Diagramm des jeweiligen X-in-the-Loop-Verfahrens in allen Dimensionen mit dem Kiviat-Diagramm des Testfalls überdeckt. Dadurch wird gezeigt, dass die vom Testfall geforderten Diskretisierungsstufen vom X-in-the-Loop-Verfahren erfüllt werden. Im zweiten Schritt werden die Kiviat-Diagramme der geeigneten Verfahren erweitert. Dazu wird das Diagramm um eine z-Achse ergänzt, auf der die Güte der verfügbaren Modelle, die Einsatzkosten, der Zeiteinsatz und weitere Bewertungskriterien für die jeweiligen Diskretisierungsstufen aufgetragen werden. Abbildung 9.3 zeigt exemplarisch die Erweiterung eines Kiviat-Diagramm um die z-Achse, auf der eine beispielhafte Gütefunktion für die Fahrdynamik aufgetragen wurde. Weitere Bewertungsdimensionen können aufgrund der Darstellungsform von Kiviat-Diagrammen flexibel hinzugefügt werden.



**Abbildung 9.3:** Unausgefülltes Kiviat-Diagramm mit den Stufen: 1 = simuliert, 2 = emuliert, 3 = real vorhanden und einer eingetragenen Gütefunktion für die Fahrdynamik auf der z-Achse

Anschließend werden die Testfälle anhand einer Bewertungsfunktion den X-in-the-Loop-Verfahren zugeordnet. Die Zuordnung eines Testfalls kann beispielsweise durch eine gewichtete Bewertungsfunktion für die verschiedenen Bewertungskriterien dargestellt werden. Dadurch wird jedem Testfall in Bezug auf die Anforderungen mit Hilfe der Bewertungsfunktion ein X-in-the-Loop-Verfahren systematisch zugeordnet. Für das Beispiel einer gewichteten Bewertungsfunktion ergibt sich die Gesamtgüte  $G$  aus der Summe der gewichteten Gütefunktionale, dargestellt in den Gleichungen 9.1 und 9.2.

$$G = \sum_{n=0}^N a_n \sum_{j=0}^J k_{ijn} \quad (9.1)$$

$$\sum_{n=0}^N a_n = 1 \quad (9.2)$$

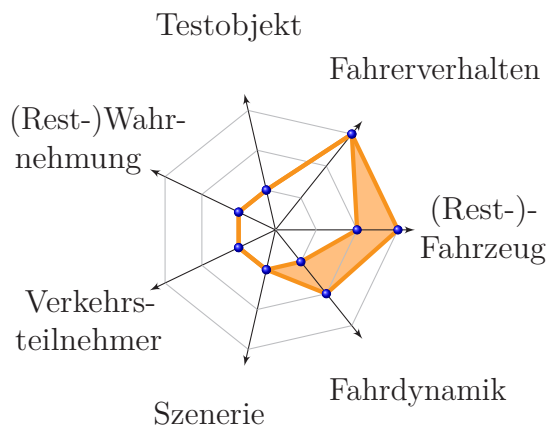
Dabei beschreibt  $a_n$  den Gewichtungsfaktor für das Bewertungskriterium  $n$  und  $k_{ijn}$  die Güte von Dimension  $j$  auf Stufe  $i$  für das Bewertungskriterium  $n$ .

### 9.3.2 Beispiel für die Zuordnung eines Testfalls auf X-in-the-Loop-Verfahren

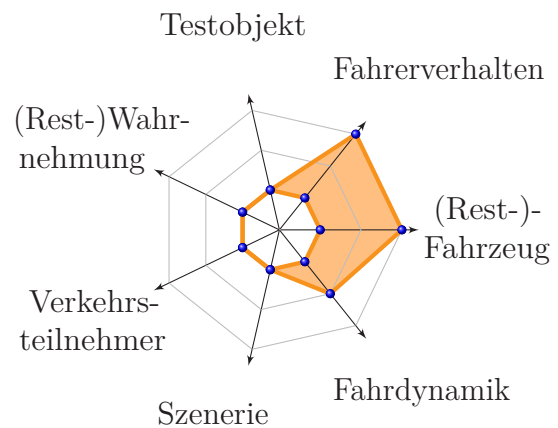
Die Methode wird im Folgenden exemplarisch anhand der Fahrfunktion aus Kapitel 3.3 angewendet. Die Fahrfunktion übernimmt die Querführung des Fahrzeugs. Während der Tests soll das Modul der Merkmalsextraktion und modellbasierten Filterung sowie das Modul der Stabilisierung getestet werden. Wie in Kapitel 3.3 beschrieben, besteht die Möglichkeit, die Fahrfunktion in ein Gesamtsystem mit Automatisierungsgrad 2 - 5 zu integrieren. Die sich für die Zuordnung ergebenden Anforderungen aus den verschiedenen Automatisierungsgraden werden im Folgenden diskutiert.

Bei einer Integration der Fahrfunktion in ein Gesamtsystem mit Automatisierungsgrad 2 - 4 besteht die Möglichkeit, dass der Fahrer die Querführung während der Fahrt übernimmt. Daher müssen für diese Automatisierungsgrade Übergabeszenarien getestet werden. Dadurch besteht die Forderung, dass eine potentielle Übergabe an einen realen Fahrer möglich ist, wodurch ein reales Fahrerverhalten zur Verfügung gestellt werden muss. Bei einer Integration in ein Gesamtsystem mit Automatisierungsgrad 5 findet keine Übergabe an den Fahrer statt, wodurch bei diesen Tests kein Fahrerverhalten benötigt wird. Somit muss das Fahrerverhalten bei der Auswahl eines X-in-the-Loop-Verfahrens nicht beachtet werden, wodurch vom Testfall keine expliziten Diskretisierungsstufen gefordert werden. Bei einem Test des Systems mit Automatisierungsgrad 2 - 4 muss das (Rest-)Fahrzeug emuliert werden oder real vorhanden sein, da eine Mensch-Maschine-Schnittstelle, wie ein Lenkrad, benötigt wird, um die potentielle Übergabe der Fahrzeugführung an den Fahrer zu testen. Für das Gesamtsystem mit Automatisierungsgrad 5 wird keine Übergabe an den Fahrer getestet, wodurch die Mensch-Maschine-Schnittstelle nicht benötigt wird. Dadurch wird, wie beim Fahrerverhalten, keine explizite Diskretisierungsstufe gefordert. Für die Merkmalsextraktion und modellbasierte Filterung wird eine wahrgenommene Szene benötigt. Um in einer kurzen Zeitspanne eine hohe Variation von verschiedenen Szenarien zu testen, sollen die Szenerie und die übrigen Verkehrsteilnehmer simuliert werden. Daher besteht die Forderung, dass auch die (Rest-)Wahrnehmungskomponenten simuliert werden. Dies gilt für die Automatisierungsgrade 2 - 5. Für die Fahrdynamik wird gefordert, dass diese

simuliert oder emuliert werden kann, um die Fahrfunktionen für verschiedene Fahrzeugtypen zu testen. Die Abbildungen 9.4 und 9.5 zeigen die Anforderungen der Testfälle an die X-in-the-Loop-Verfahren.



**Abbildung 9.4:** Anforderungen der Testfälle bei einer Integration in ein Gesamtsystem mit Automatisierungsgrad 2 - 4



**Abbildung 9.5:** Anforderungen der Testfälle bei einer Integration in ein Gesamtsystem mit Automatisierungsgrad 5

Durch einen visuellen Vergleich der oben abgebildeten Kiviat-Diagramme mit den Kiviat-Diagrammen der klassifizierten X-in-the-Loop-Verfahren aus Kapitel 4.3.3 lassen sich ungeeignete und geeignete Verfahren für die Testdurchführung identifizieren. Für das System mit Automatisierungsgrad 2 - 4 klassifizieren sich die Verfahren *Driver-in-the-Loop* mit einem stationären oder beweglichen Simulator sowie das Verfahren *Vehicle-Hardware-in-the-Loop*. Für ein System mit Automatisierungsgrad 5 kann zusätzlich das Verfahren *Software-in-the-Loop* verwendet werden, da kein reales Fahrerverhalten benötigt wird. Dies gilt ebenso für ein System mit Automatisierungsgrad 4, wenn keine Übergabe an den Fahrer getestet werden soll.

Für eine weitere Selektion der geeigneten Verfahren wird exemplarisch die zweite Stufe der Zuordnungsmethode auf das Gesamtsystem mit Automatisierungsgrad 5 angewendet. Dazu werden die Bewertungskriterien Zeiteinsatz und Einsatzkosten verwendet. Mit der Voraussetzung, dass das Testobjekt die Berechnungen schneller als in Echtzeit durchführen kann, wird beim *Software-in-the-Loop*-Verfahren eine Testdurchführung schneller als in Echtzeit ermöglicht. Bei den übrigen Verfahren ist die Durchführung der Testfälle lediglich in Echtzeit möglich. Dadurch hat das *Software-in-the-Loop*-Verfahren gegenüber den übrigen Verfahren einen geringeren Zeitbedarf. Des Weiteren wird an dieser Stelle angenommen, dass die Durchführung eines Testfalls im *Software-in-the-Loop*-Verfahren mit geringeren Kosten verbunden ist als bei der Durchführung eines Testfalls im *Driver-in-the-Loop*-Verfahren oder *Vehicle-Hardware-in-the-Loop*-Verfahren. Dies kann unter anderem durch die Kosten der zusätzlich benötigten Hardware in Form von Simulatoren begründet werden. Nach der Anwendung der zweiten Stufe der Zuordnungsmethode wurde das *Software-in-the-Loop*-Verfahren nach den Bewertungskriterien Kosten und Zeitaufwand als das am besten geeignete Verfahren für die Testdurchführung identifiziert.



### 9.3.3 Zusammenfassung

In diesem Kapitel wurde eine zweistufige Methode für die Zuordnung von Testfällen auf X-in-the-Loop-Verfahren vorgestellt. Auf Basis der Klassifikation der X-in-the-Loop-Verfahren aus Kapitel 4.3.3 und der Anforderungen eines Testfalls an die X-in-the-Loop-Verfahren werden in der ersten Stufe des Verfahrens geeignete X-in-the-Loop-Verfahren für die Testdurchführung identifiziert. Dies wurde durch die exemplarische Anwendung der Zuordnungsmethode auf die aktive Querführungsfunktion aus Kapitel 3.3 demonstriert.

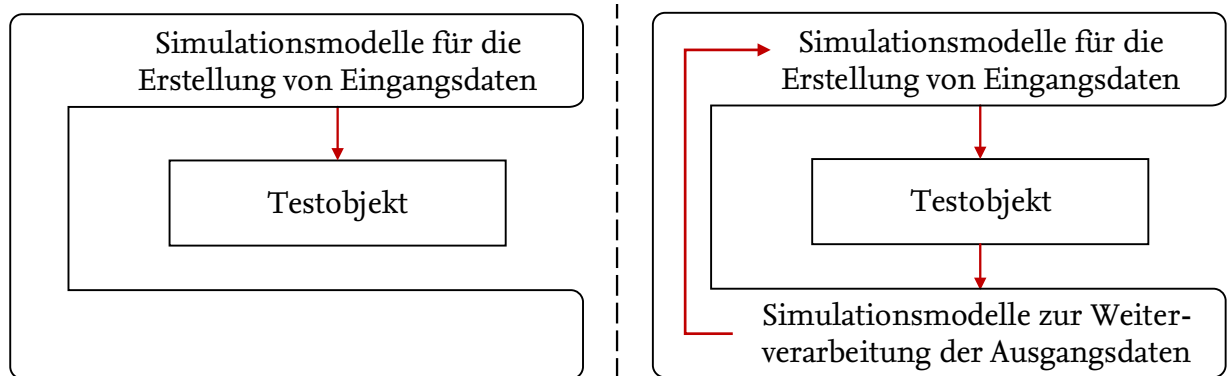
Für die zweite Stufe der Methode werden Informationen über die Kosten und den Zeitaufwand für die Testdurchführung benötigt. Weiterhin sind Informationen über die Güte der verfügbaren Simulationsmodelle und Emulationen notwendig. Diese Informationen haben basierend auf der verwendeten Bewertungsfunktion einen bedeutenden Einfluss auf die Entscheidung für oder gegen ein Verfahren. Die Bewertung der Güte von Simulationsmodellen ist derzeit eine Herausforderung, da unter anderem Metriken für die Bewertung der verschiedenen Modelle fehlen. Ohne die Bewertungen der Modelle besteht derzeit nur die Möglichkeit, mit der Zuordnungsmethode die nach den Anforderungen des Tests geeigneten Verfahren anhand der Kosten und des Zeitaufwands zu bewerten. Sobald Metriken für die Bewertung von Simulationsmodellen und Emulationen verfügbar sind, bietet die vorgestellte Methode das Potential, die Testfälle systematisch anhand der Modellgüte auf die X-in-the-Loop-Verfahren zuzuordnen.

## 9.4 Testtreiber für die Module der funktionalen Systemarchitektur

Das folgende Unterkapitel identifiziert Testtreiber, die für den Test der Module der funktionalen Systemarchitektur aus Kapitel 3.2 benötigt werden. Ein Testtreiber ist nach IEEE 610 (1990) und Hamburg (2015) eine Softwarekomponente oder ein Testwerkzeug, welches eine zu testende Komponente beziehungsweise System aufruft und/oder steuert und zusätzlich die Ausführung überwacht. Ein Testtreiber muss somit auch die Eingangsdaten für das Testobjekt erstellen. Bei einer Testdurchführung mit Rückführung der Ausgaben des Testobjekts muss der Testtreiber zusätzlich die Ausgangsdaten des Testobjekts verarbeiten und diese gegebenenfalls speichern, um aus den Ausgangsdaten neue Eingangsdaten für das Testobjekt zu erstellen. Abbildung 9.6 zeigt schematisch einen Testtreiber für die Testdurchführung ohne und mit Rückführung der Ausgaben des Testobjekts.

Im Folgenden werden die Herausforderungen identifiziert, die sich bei der Erstellung von Testtreibern aus der Simulation beziehungsweise Emulation für die verschiedenen Module der funktionalen Systemarchitektur ergeben. Dazu werden zunächst für jedes Modul die Eingangs- als auch Ausgangsdaten beschrieben, um anschließend die benötigten Testtreiber zu identifizieren. Dazu werden die Simulationsmodelle beschrieben, die für die Erstellung der Eingangsdaten benötigt werden. Die Reihenfolge der Unterkapitel folgt dem Aufbau der vorgestellten funktionalen Systemarchitektur.

Weiterhin wird analysiert, inwiefern die Module mit Hilfe der Simulation testbar beziehungsweise bewertbar sind. Ein Modul ist in der vorliegenden Arbeit mit Hilfe der Simulation testbar und bewertbar, wenn die Möglichkeit besteht, die genauen Eingangs- als auch Ausgangsdaten für das jeweilige Modul mit Hilfe der Simulation zu erstellen. In diesem Fall können die Ausgangsdaten des Moduls direkt mit den genauen Ausgangsdaten aus der



**Abbildung 9.6:** Schematische Darstellung eines Testtreibers ohne (links) und mit (rechts) Rückführung der Ausgaben des Testobjekts

Simulation verglichen werden. Wenn lediglich die Eingangsdaten erzeugt werden können, ist das Modul mit Hilfe der Simulation testbar jedoch nicht bewertbar, da die genauen Ausgangsdaten fehlen. In diesem Fall müssen weitere von der Simulation unabhängige Bewertungskriterien, wie sie in Kapitel 10 vorgestellt werden, zur Bewertung hinzugezogen werden.

### 9.4.1 Sensorik

Im Folgenden wird lediglich die Umfeldsensorik der Architektur aus Kapitel 3.2 betrachtet. Die Fahrzeugsensorik wird in der vorliegenden Arbeit nicht weiter analysiert.

Um einen Testtreiber für ein System inklusive der Empfangseinheit der Umfeldsensorik bereitzustellen, werden ein Umfeldmodell und ein Emitter benötigt, welche die Eingangsgrößen für die Umfeldsensorik zur Verfügung stellen. Ein möglicher Emitter für Radarsignale wird von Rohde & Schwarz (2015) vorgestellt. Nach der Zuordnungsmethode stellt der Test inklusive Emitter eine Emulation dar, da die Eingangssignale durch eine hardwareähnliche Einheit erstellt werden. Die Informationen über die auszusendenden Größen werden über ein Umfeldmodell berechnet.

Die Umfeldsensoren werden in der vorliegenden Arbeit in die Gruppe der aktiven und passiven Umfeldsensoren unterteilt. Aktive Umfeldsensoren senden ein Signal aus, welches über Reflexionen zur Empfangseinheit des Umfeldsensors gelangt und dort ausgewertet wird. Passive Umfeldsensoren empfangen die Energie, beispielsweise in Form von Licht oder Wärmestrahlung, die von den Elementen des Umfelds abgestrahlt wird, ohne die Elemente aktiv zu bestrahlen. Zur Gruppe der aktiven Umfeldsensoren gehören unter anderen Radar<sup>13</sup>, Lidar<sup>14</sup>, Ultraschall-Sensoren sowie Infrarotkameras im Nahinfrarotbereich. Beispiele für passive Umfeldsensoren sind Kameras im Sicht- oder im Ferninfrarotbereich.

Je nach Sensortechnologie ergeben sich für den Testtreiber inklusive des Emitters unterschiedliche Anforderungen. Die Anforderungen an einen Testtreiber für aktive Umfeldsensoren ergeben sich durch die benötigte Modulation der Sendeeinheit und durch die Berechnung der Umfeldeffekte für die ausgesendeten Signale, welche vom Umfeldsensor über die reflektierenden Elemente zurück zur Empfangseinheit des Umfeldsensors betrachtet werden müssen.

<sup>13</sup>Abkürzung für Radio detection and ranging. Frei übersetzt: Funkortung und -abstandsmessung

<sup>14</sup>Abkürzung für Light detection and ranging.

Beim Radar müssen beispielsweise Effekte wie Fluktuationen und Mehrwegeausbreitungen beachtet werden. Nach Meinecke (2016) können diese Effekte zum Teil selbst mit statistischen Methoden nur ansatzweise beschrieben werden.

Trotz der hohen Anforderungen sind Ansätze für Testtreiber vorhanden, die eine Integration der Empfangseinheit der Umfeldsensorik in die Testschleife ermöglichen. Ein Ansatz zur Integration von Ultraschallsensoren wird von Pfeffer u. a. (2015) vorgestellt. Die Ultraschallsensoren werden in einer Box verbaut und mit einem Emitter stimuliert, welcher seine Eingangsdaten in Form von simulierten Abstandswerten erhält. Petit (2015) stellt einen Ansatz für die Emulation der Eingangsdaten von Laserscannern vor. Die Ansätze zeigen, dass es grundsätzlich möglich ist, die Umfeldsensorik mit simulierten Daten zu stimulieren, wenn ein Emitter verfügbar ist. Jedoch zeigen die Ansätze zugleich die Herausforderungen für solche Testtreiber. Beispielsweise ist für Ultraschallsensoren der Wind ein entscheidender Umwelteinfluss, der in der von Pfeffer u. a. (2015) vorgestellten Box nicht emuliert wird. Die sich aus dem Wind ergebenden Signale für die Ultraschallsensoren müssen somit vom Umfeldmodell dargestellt werden. Für eine realitätsnahe Simulation beziehungsweise Emulation der Eingangsgrößen des Umfeldsensors wird demnach ein genaues<sup>15</sup> modelliertes Umfeldmodell benötigt, welches die Umfeldeffekte auf Basis von physikalischen Prinzipien berücksichtigt.

Für die passiven Umfeldsensoren ergeben sich die Anforderungen an den Testtreiber durch die Erstellung der Eingangssignale, welche von der Energiequelle zur Empfangseinheit übertragen werden. Um die berechneten Signale zur Empfangseinheit des Umfeldsensors zu übertragen, wird ebenfalls ein Emitter benötigt. Für eine Kamera besteht die Möglichkeit, einen Bildschirm als Emitter zu verwenden. Dieser stellt die synthetisch erzeugten Bilder dar, welche von der Kamera aufgenommen werden. Dieser Ansatz wird beispielsweise von Burdessa u. a. (2010) vorgestellt. Schmidt (2012) und Nentwig (2013) zeigen in ihren Arbeiten, wie synthetische Bilder für den Test von Assistenzsystemen einsetzbar sind. Jedoch zeigen die Arbeiten, dass bei komplexeren Szenarien, bei denen viele Elemente miteinander interagieren, wie beispielsweise Gegenlicht von verschiedenen Lichtquellen sowie Reflexionen von Lichtern, die Modelle immer wieder für jedes spezielle Szenario parametrisiert und validiert werden müssen, was einen erheblichen Aufwand bedeuten kann (Nentwig, 2013, S.201).

Die Einkopplung der Emulation vor der Umfeldsensorik hat den Vorteil, dass sich die Empfangseinheit der Umfeldsensorik in der Testschleife befindet. Damit wird die Möglichkeit geschaffen, Szenarien zu testen, die aufgrund von Randbedingungen nicht in der Realität durchführbar sind. Insgesamt zeigt sich, dass die Erstellung von Testtreibern für die Erstellung von realitätsnahen Eingangsdaten, wie sie von der Empfangseinheit erwartet werden, aufgrund der Notwendigkeit von umfassenden Umfeldmodellen inklusive Effektmodellen nur mit erheblichem Aufwand möglich ist. Dies gilt besonders bei aktiven Umfeldsensoren. Weiterhin werden die passenden Emitter für die Emulation der Eingangsdaten benötigt. Aus diesen Gründen ist derzeit im Konzept des modularen virtuellen Testbaukastens kein Testtreiber für diese Art der Tests vorgesehen.

### 9.4.2 Kommunikation

Im Folgenden werden Testtreiber für die Kommunikation zwischen der automatisierten Fahrfunktion und den übrigen Verkehrsteilnehmern sowie den Insassen innerhalb eines

<sup>15</sup>Genau bedeutet, dass die Daten nach DIN 5725 (2002) eine hohe Richtigkeit und Präzision haben.

automatisierten Fahrzeugs betrachtet. Die Kommunikation zu anderen Verkehrsteilnehmern findet nach Matthaei (2015, S.56) auf optischer und akustischer Ebene statt. Als optische Kommunikation wird die Aktivierung des Fahrtrichtungsanzeiger oder der Bremslichter verstanden. Dies wird von Matthaei u. a. (2015) durch die direkte Kommunikation mit Menschen, wie Gesten von Fußgängern oder Polizisten, und die Körpersprache von Tieren ergänzt. Als akustische Kommunikationsebene sieht Matthaei (2015, S.56) die Aktivierung der Hupe.

Für eine optische, bilaterale Kommunikation zwischen der automatisierten Fahrfunktion und den Komponenten der Simulation wird ein Umfeldmodell benötigt. Dieses beinhaltet die Verkehrsteilnehmer, die zum einen Kommunikationssignale aussenden und zum anderen die Kommunikationssignale der Fahrfunktion aufnehmen und verarbeiten. Für eine optische Kommunikation, die von der Fahrfunktion durch Kameras verschiedener Arten aufgenommen wird, werden synthetische Bilder der Simulation benötigt, auf denen Personen, Fahrzeuge oder Tiere durch Bewegungen kommunizieren. Um die optische Kommunikation der Fahrfunktion in das Umfeldmodell zu übertragen, können die Bussysteme des Fahrzeugs verwendet werden, auf denen die Signale zur Steuerung der optischen Aktoren gesendet werden. Das Umfeldmodell muss die Kommunikation der Fahrfunktion verstehen und interpretieren können, um das eigene Verhalten darauf anzupassen. Dafür werden Interpretationsmodelle für eine realitätsnahe Kommunikationen benötigt, da in der realen Kommunikation Signale nach Rupprecht (2014, S.41f) womöglich nicht korrekt empfangen oder fehlinterpretiert werden. Weiterhin ist die regional verschiedene Bedeutung von optischen Gesten in den Interpretationsmodellen zu beachten. Bei akustischen Signalen ist dies äquivalent, mit dem Unterschied, dass ein akustischer Emitter benötigt wird, um die akustischen Sensoren der Fahrfunktion zu stimulieren. Um eine Kommunikation mit dem Fahrer zu testen, werden zusätzlich Fahrermodelle benötigt, die die optischen und akustischen Signale der Fahrfunktion interpretieren und darauf reagieren.

Für einen Testtreiber, der einen Test der Kommunikation inklusive der Kommunikationsempfänger für Fahrzeug-zu-X-Kommunikation ermöglicht, wird ein Emitter benötigt, der die Signale an die Empfangseinheit der Kommunikationsempfänger überträgt. Für die Berechnung der auszusendenden Signale wird ein Umfeldmodell auf Basis von physikalischen Prinzipien benötigt, welches für eine realitätsnahe Abbildung der Eingangssignale Umfeldeffekte, wie die Abschattung der Signale durch Gebäude, Laufzeitdifferenzen und Mehrwegeausbreitungen, modelliert. Daher stellt der Test der Kommunikationssensorik hohe Anforderungen an das Umfeldmodell, da genaue Modelle für die Berechnung der Signalausbreitungen im Raum benötigt werden, um realitätsnahe Sendedaten für die Emitter der Signale zu berechnen. Weiterhin fehlen Metriken, um die Güte der Modelle zu bewerten und eine Validierung der Modelle durchzuführen. Daher wird diese Art der Tests momentan im Konzept des modularen virtuellen Testbaukastens nicht weiterverfolgt.

Eine Ausnahme stellt die optische Kommunikation dar. Aufgrund der vorhandenen Emitter und der Forschung und Entwicklung von Computergrafiken besteht die Möglichkeit, auf Bewegungsmodelle von Menschen und Tieren zurückzugreifen. Diese können über synthetisch erstellte Bilder einer Kamera vorgespielt werden. Die Firma *DiGuy*<sup>16</sup> stellt beispielsweise eine realitätsnahe Fußgängersimulation vor. Daher ist der Test von optischer Kommunikation

---

<sup>16</sup><http://www.mak.com/products/humans>, abgerufen: 07.03.2016

mit Hilfe der Simulation durchaus denkbar, wird jedoch derzeit im Testbaukasten nicht unterstützt.

### 9.4.3 Lokalisierung und Karten

Um einen Testtreiber für die Tests der Module innerhalb der Spalte Lokalisierung und Karten bereitzustellen, wird eine Simulation der Lokalisierungssensoren und der Umfeldsensorik benötigt. Dazu sind Simulationsmodelle notwendig, die die Ausgangsdaten der Sensoren zur Verfügung stellen. Im Gegensatz zu den zuvor beschriebenen Testtreibern werden sowohl die Sende- als auch die Empfangseinheit der Sensorik simuliert. Damit ergeben sich an den Testtreiber insgesamt geringere Anforderungen, da die bereits vorverarbeiteten Ausgangssignale der Lokalisierungs- und Umfeldsensorik modelliert werden und nicht die physikalischen Signale für die Empfangseinheit.

Die Lokalisierungssensoren stellen die Pose des Fahrzeugs zur Verfügung. Für die Berechnung der Ausgangsdaten der Umfeldsensoren werden verschiedene Arten von Simulationsmodellen benötigt, die die Daten in verschiedenen Gütestufen ausgeben. Für die Definition der Güte wird auf die in Kapitel 9.1.1 eingeführten Gütemaße zurückgegriffen.

Für den Test der Module werden sowohl genaue als auch fehlerbehaftete beziehungsweise realitätsnahe Daten benötigt. Die genauen Daten werden im Testbaukasten direkt aus der Simulation übernommen. Eine erste Möglichkeit, die Modelldaten den realen Sensordaten anzunähern, bieten Fehlermodelle in Form von Rauschmodellen. Levinson u. a. (2007) gehen beispielsweise davon aus, dass die Signale von GNSS<sup>17</sup>-Empfängern einem systematischen Rauschen unterliegen. Mittels Rauschmodellen lässt sich die Qualität der Daten um eine gewisse Länge in die Richtung realitätsnaher Daten verschieben. Die Länge ist allerdings aufgrund von fehlenden Metriken nicht messbar. Für eine weitere Verschiebung in Richtung realitätsnaher Daten sind detailliertere Modellierungen des Umfelds notwendig. Diese sind derzeit im modularen virtuellen Testbaukasten nicht realisiert, sodass aktuell nur einfache Rauschmodelle für den Test der Spalte Lokalisierung und Karten anwendbar sind.

### 9.4.4 Merkmalsextraktion und modellbasierte Filterung

Um einen Testtreiber für das Modul der Merkmalsextraktion und modellbasierten Filterung zu realisieren, werden die Ausgangsdaten der Umfeldsensorik sowie der Lokalisierung und Karten auf mikroskopischer Ebene aus der Simulation benötigt. Durch eine Erstellung der Ergebnisse der Merkmalsextraktion durch die Simulation besteht die Möglichkeit, den Test der Merkmalsextraktion durch den Testtreiber zu überspringen, da die bereits vorverarbeiteten Informationen zur Verfügung gestellt werden. Ein Testtreiber für die Selbstwahrnehmung wird in der vorliegenden Arbeit nicht betrachtet, da das Konzept des Testbaukastens dies im derzeitigen Stand nicht vorsieht. Eine Erweiterung des Testbaukastens in weiteren Forschungsarbeiten ist aber denkbar.

Um die Güte der Module zu bewerten, werden zusätzlich die genauen Ausgangsdaten aus der Simulation benötigt. Diese sind unter anderem die Informationen über die dynamischen Elemente um das Testobjekt, eine genaue stationäre Belegungskarte, Informationen über die Fahrstreifen sowie der Zustand von Lichtsignalanlagen und die Position und Bedeutung von Verkehrszeichen. Um perfekte Referenzdaten zu erstellen, müssen die Informationen

<sup>17</sup>GNSS: Globales Navigationssatellitensystem



über die Elemente, die sich im Sichtbereich der Umfeldsensoren befinden, genau sein, d.h. das die Daten nach DIN 5725 (2002) eine hohe Richtigkeit und Präzision haben.

Für die Erstellung der Ausgangsdaten der Lokalisierung wird eine hochgenaue Pose auf mikroskopischer Ebene benötigt. Diese wird im Testbaukasten durch ein perfektes Modell aus der submikroskopischen Simulation bestimmt. Für nicht perfekte Positionsdaten werden Rauschmodelle verwendet. Des Weiteren wird eine hochgenaue Karte benötigt. Diese wird im modularen virtuellen Testbaukasten durch das *OpenDrive*-Format<sup>18</sup> beschrieben, welches von Dupuis u. a. (2010b) vorgestellt wurde und eine submikroskopische Beschreibung der Straße ermöglicht.

Um die Ausgangsdaten der Umfeldsensoren zu erstellen, werden die Sende- und Empfangseinheit des Sensors simuliert, wodurch die Simulationsmodelle die Messdaten des Sensors ausgeben. Im Folgenden werden die Sensorarten Lidar, Radar und Kamera betrachtet.

### Radar

Ein Radar misst je nach Modulationsverfahren den radialen Abstand und/oder die relative Geschwindigkeit zu detektierten Objekten. Aus einer Anordnung von mehreren Erfassungsbereichen (Sensorkulen) wird der Azimutwinkel des Objekts geschätzt. Winner (2015b) gibt einen ausführlichen Überblick über verschiedene Modulationsverfahren der Radarsensorik. Für eine vereinfachte Simulation eines Radars besteht die Möglichkeit ein Strahlenmodell einzusetzen. Diese Möglichkeit wird derzeit im Testbaukasten verwendet. Ein ähnliches Prinzip wird von Cao u. a. (2015) als Gray-Box Modell eines Radarsensors vorgeschlagen.

Das Strahlenmodell sendet virtuelle Strahlen vom Sender über die Breite des Erfassungsbereichs. Damit werden perfekte Treffpunkte des Radars simuliert. Die Informationen über den Treffpunkt werden dem Modul der Merkmalsextraktion und modellbasierten Filterung übergeben. Mit diesem simplen Modell werden der Erfassungsbereich des Umfeldsensors und die Verdeckungen von Objekten berechnet. Eine Möglichkeit, die perfekten Daten in Richtung realitätsnaher Daten zu verschieben, ist nach Cao u. a. (2015) und Lackmann (2015) ein nachträgliches Verrauschen der Informationen über den Treffpunkt. Dies stellt ein stark vereinfachtes Fehlermodell dar. Fehlermodelle, die realitätsnähere Daten zur Verfügung stellen, benötigen eine detailliertere Modellierung des Umfelds. Dies bedeutet, dass unter anderem die materiellen Eigenschaften der Oberflächen der Elemente und die Eigenschaften des Wetters modelliert werden, da die Empfangsdaten stark von diesen Umgebungsbedingungen abhängen. Gotzig und Geduld (2015, S.318) zeigen beispielsweise die Dämpfung von Signalen über verschiedene Frequenzbereiche und Wettersituationen. Eine Umsetzung eines erweiterten Fehlermodells, welches prinzipbedingte Fehler abdeckt, wird von Lackmann (2015) als Erweiterung für das Strahlenmodell vorgestellt. In diesem Fehlermodell werden die Rückstrahlwerte in Bezug zum Wetter berechnet. Weiterhin wird eine Approximation für die Mehrwegeausbreitungen und atmosphärischen Dämpfungen der Signale für ein Radar implementiert. Dieses Modell wird derzeit im Testbaukasten verwendet. Weitere Implementierungen sind nicht umgesetzt.

In der Merkmalsextraktion werden aus den Treffpunkten der Radarstrahlen Objekthypothesen gebildet. Diese werden für die Nachverfolgung von Objekten verwendet. Aus der Simulation werden für die Erstellung von Objekthypothesen Modelle benötigt, die aus existierenden genauen Objektlisten Objekthypothesen erstellen. Perfekte Objekthypothesen

<sup>18</sup><http://www.opendrive.org>, abgerufen: 07.03.2016



werden direkt aus den perfekten Objektlisten generiert. Für die Erstellung von realitätsnahen Objekthypothesen werden prinzipbedingte Fehlermodelle benötigt. Erste Fehlermodelle für die Veränderung von Objektlisten werden von Lackmann (2015) vorgestellt. In der Arbeit werden verschiedene Rausch- und Detektionsmodelle vorgestellt, um die Objektlisten zu manipulieren.

### Lidar

Ein Lidar beziehungsweise Laserscanner misst die Entfernung zu Objekten und je nach Auflösung, d.h. Strahlen pro Grad in horizontaler und vertikaler Richtung, den Azimut- und Elevationswinkel zu den Objekten. Zusätzlich ist es möglich, den Reflexionsgrad des Auftreffpunkts zu erfassen. Für die Erstellung dieser Eingangsdaten wird im Testbaukasten ebenfalls das oben erwähnte Strahlenmodell verwendet. Mit dem Modell werden sowohl 2D- als auch 3D-Strahlen versendet, um die Laserstrahlen zu simulieren. Ein ähnliches Modell zur Erstellung von Laserrohdaten wird ebenfalls von Berger (2010, Kap.6.4.6) verwendet. Weiterhin bietet das Modell die Möglichkeit, die Strahlen eines sich rotierenden Umfeldsensors zu simulieren, bei dem die einzelnen Strahlen zu verschiedenen Zeitpunkten verschickt werden. Die Auswirkungen dieses Effekts werden beispielsweise von Rieken und Maurer (2016) untersucht. Im aktuellen Strahlenmodell werden derzeit nur perfekte Strahlen simuliert. Dies bedeutet, dass pro Strahl nur das dichteste Objekt getroffen wird und keine Mehrfachziele pro Strahl möglich sind. Des Weiteren werden derzeit keine Reflexionen oder Brechungen der Laserstrahlen berechnet. Hierfür sind Erweiterungen des Strahlenmodells notwendig.

Die in der Merkmalsextraktion der Laserrohdaten generierten Informationen werden, wie von Matthaei (2015, Kap.11) gezeigt, in den Modulen der Fahrstreifenverfolgung genutzt. Eine weitere Anwendung ist die von Rieken u. a. (2015) gezeigte Klassifikation von stationären und dynamischen Elementen. Dabei werden stationäre Elemente in eine Belegungskarte eingetragen und dynamische Elemente für die Objektverfolgung verwendet. Die Daten der Merkmalsextraktion werden im Testbaukasten basierend auf dem existierenden Strahlenmodell und den zur Verfügung stehenden Objektlisten aus der Simulation generiert. Auf der Basis von perfekten Objektlisten werden die Informationen über den Objekttyp und deren Geschwindigkeit generiert. Damit besteht die Möglichkeit, die Dynamik- und die Objektklassifikation, die in der Merkmalsextraktion stattfindet, mit Hilfe der Simulation zu überspringen. Die Informationen über die Fahrstreifenmarkierungen für die Fahrstreifenverfolgung werden aus dem *OpenDrive*-Format extrahiert.

Eine erste Umsetzung eines Fehlermodells für die Sensordaten und die Daten der Merkmalsextraktion wurde in der Arbeit von Lackmann (2015) umgesetzt. Dort wurden die Effekte der Verdeckung, des Rauschens sowie der Dämpfung für das Strahlenmodell und die Objektlisten umgesetzt. Weitere Effekte sind derzeit im Testbaukasten nicht umgesetzt.

### Kamera

Für die Erstellung von Ausgangsdaten einer Kamera werden Kameramodelle benötigt, die synthetische Bilder generieren. Hierzu wird ein Bildgenerator, wie der *ImageGenerator* der Software *Virtual Test Drive* von Vires<sup>19</sup>, benötigt. Für die Erstellung von Bildern einer Stereo-Kamera werden zwei Instanzen der Modelle benötigt, welche die Bilder zum selben Zeitpunkt erstellen. Das Kameramodell wird mit der Pose und den intrinsischen

<sup>19</sup><http://vires.com>, abgerufen: 08.03.2016

Parametern der Kamera parametrisiert. Des Weiteren besteht die Möglichkeit, dass spezielle Effekte, wie Linsenverzerrungen, dem Bild hinzugefügt werden, um die Bilder verschiedenster Kamerasysteme zu simulieren. Dies wird beispielsweise für die Simulation eines 360° Kamerasystems benötigt, bei dem Fischaugenobjektive eingesetzt werden. Die Bilder der Kamera werden für die Erkennung von Verkehrszeichen und Lichtsignalanlagen, für die Verfolgung von Fahrstreifen, für die Befüllung von stationären Belegungskarten und die Verfolgung von dynamischen Elementen verwendet. Ein Framework für die Erstellung von synthetischen Kamerabildern wird von Mueller u. a. (2015) vorgestellt. Das Framework unterstützt verschiedene Wettereffekte, wie Sonne, Regen, Schnee oder Nebel.

Die Merkmalsextraktion erzeugt Informationen über die Fahrstreifen, die erkannten Verkehrszeichen und Lichtsignalanlagen, die Freibereiche, die dynamischen Elemente sowie für Stereo-Kameras ein Flußbild, Disparitätsbild und ein Tiefenbild. Rohdaten, d.h. Bilder in verschiedensten Formen, werden derzeit mit dem Testbaukasten nicht erstellt und ausgewertet. Aktuell besteht die Möglichkeit die in der Simulation erstellten Bilder an die Verarbeitungskette ohne Änderungen weiterzugeben. Die Verzerrung von Bildern wird derzeit nicht unterstützt.

Informationen über die Breite und die Farbe der Fahrstreifen sowie den Verlauf in Form von Klothoidenparametern oder Punktlisiten werden im Testbaukasten aus dem *Open-Drive*-Format extrahiert. Verkehrszeichen und bewegliche Elemente werden per Objektliste empfangen und durch das Strahlenmodell auf Verdeckung überprüft. Dabei handelt es sich um perfekte Daten, die mit dem Framework von Lackmann (2015) manipulierbar sind. Weitere Modelle für eine realitätsnähere Abbildung von Bildern sind derzeit im Testbaukasten nicht vorhanden.

### **Zwischenfazit zur Merkmalsextraktion und modellbasierten Filterung**

Nach der Betrachtung der Testtreiber für die Merkmalsextraktion und der modellbasierten Filterung wird zusammengefasst, dass die Erstellung von realitätsnahen Sensorrohdaten eine Herausforderung darstellt. Erste Fehlermodelle ermöglichen eine Simulation von prinzipbedingten Fehlern, um die perfekten Daten in Richtung realitätsnaher Daten zu verschieben. Jedoch kann mit den Fehlermodellen nicht die komplette Distanz zwischen den Modelldaten und den Daten, die eine reale Umfeldsensorik unter den Umfeldbedingungen messen würde, zurückgelegt werden. Um die Distanz zwischen den realen und simulierten Daten weiter zu verkürzen, sind nach Roth (2015) und dem Forschungsprojekt *PEGASUS*<sup>20</sup> (Plättner u. a., 2016) weitere Forschungsarbeiten notwendig. Dabei sollte der Fokus der Forschung auf detailliertere Ansätze für Sensormodelle und Metriken zur Bestimmung der Modellgüte liegen. Somit ist die Simulation bei der Erstellung von Sensorrohdaten derzeit nur bedingt einsetzbar.

Derzeit zeigt die Simulation einen höheren Nutzen bei der Erstellung von Eingangsdaten für die modellbasierte Filterung, da sowohl die genauen Eingangs- als auch Ausgangsdaten aus der Simulation erzeugbar sind. Damit wird die Möglichkeit geschaffen, die Module zu testen und die Güte zu bestimmen, in dem die genauen Ausgangsdaten der Simulation mit den Ausgangsdaten der Module verglichen werden. Weiterhin bieten prinzipbedingte Fehlermodelle das Potential, die Qualität der Eingangsdaten zu variieren. Damit kann mit Hilfe der Simulation die grundsätzliche Funktionsweise der modellbasierten Filterung

---

<sup>20</sup>PEGASUS: Projekt zur Etablierung von generell akzeptierten Gütekriterien, Werkzeugen und Methoden sowie Szenarien und Situationen zur Freigabe hochautomatisierter Fahrfunktionen

getestet werden und grundsätzliche Fehler behoben werden. Ob eine Optimierung und Parametrisierung der Filter auf Basis der simulierten Daten möglich ist, kann derzeit nicht beantwortet werden. Bisher werden hierfür reale Messdaten benötigt.

#### 9.4.5 Szenen-/Kontextmodellierung

Für einen Testtreiber des Moduls der Szenen-/Kontextmodellierung werden die Ausgangsdaten der Merkmalsextraktion und modellbasierten Filterung, eine mesoskale Karte und eine mesoskale Lokalisierung benötigt. Die zentrale Ausgangsschnittstelle der Szenen-/Kontextmodellierung ist die subjektive Szene, d.h. die eingeschränkte Szene aus der Sicht der automatisierten Fahrfunktion beziehungsweise des Testobjekts. Zur Erstellung der Szene findet zunächst eine Modellierung der Szenerie und der dynamischen Elemente statt. Daraufhin wird das Umfeld assoziiert und die subjektive Szene modelliert.

Für die Kartenbereitstellung wird das *OpenDrive*-Format verwendet, da die mesoskopische Karte aufgrund des höheren Abstraktionsgrads aus diesem Format erstellbar ist. Die Ausgangsschnittstellen des Moduls der Merkmalsextraktion und modellbasierten Filterung werden mit Hilfe des modularen virtuellen Testbaukastens in perfekter Güte mit hoher Genauigkeit als auch mit verrauschten Daten erstellt. Da die Ausgangsdaten in Form von Szenen als objektive Szenen, d.h. genau und vollständig, in der Simulation erstellbar sind, kann dieses Modul mit dem Testbaukasten getestet und bewertet werden. Zusätzlich ist es möglich, die Simulation an diversen Zwischenschritten der Modellierung einzubauen. So sind beispielsweise die Informationen verfügbar, auf welchen Fahrstreifen sich die verschiedenen Fahrzeuge befinden, wodurch die Fahrstreifenzuordnung der Fahrzeuge mit der Simulation übersprungen werden kann.

#### 9.4.6 Straßentopologie- und Verkehrsflussmodellierung

Für einen Testtreiber des Moduls der Straßentopologie- und Verkehrsflussmodellierung werden die Szene, eine makroskopische Karte und die Position auf Straßenebene benötigt. Die Ausgangsschnittstelle wird beschrieben durch ein Straßennetzwerk inklusive Verkehrsdichteinformationen. Um diese Ausgangsdaten bereitzustellen, wird eine Identifikation der Straßentopologie und des Verkehrsflusses durchgeführt.

Für die Kartenbereitstellung wird das *OpenDrive*-Format verwendet. In diesem werden sowohl die Straßengeometrie als auch -topologie beschrieben. Mit der Information über die Anzahl der Verkehrsteilnehmer, die sich auf den jeweiligen Strecken befinden, und der Information über die Strecken aus dem *OpenDrive*-Format kann die Verkehrsdichte berechnet werden. Damit wird ein Straßennetzwerk mit einer genauen Verkehrsdichte als Ausgangsdaten bereitgestellt.

Da alle Eingangs- und Ausgangsdaten für das Modul erstellbar sind, ist der Test und eine Bewertung des Moduls mit Hilfe der Simulation möglich. Daher bietet das Konzept des Testbaukastens die Möglichkeit der Durchführung solcher Tests. Jedoch wurde der Testbaukasten in seiner Entwicklung und Konzeption bisher nicht auf makroskopischer Ebene eingesetzt. Der Nachweis für die Einsetzbarkeit des Testbaukastens als Testkonzept für das Modul der Straßentopologie- und Verkehrsflussmodellierung auf makroskopischer Ebene muss noch erbracht werden.

### 9.4.7 Navigation

Für einen Testtreiber des Moduls der Navigation wird das mit der Verkehrsdichte parametrisierte Straßennetzwerk und die Mission der Fahrfunktion, in diesem Fall das Ziel der Routenplanung, benötigt. Die Ausgangsschnittstelle ist eine straßengenaue Route, die der Führung übergeben wird.

Die Route wird nach definierten Kriterien, beispielsweise kürzeste, schnellste oder ökonomischste Route, geplant. Diese Planung ist ebenfalls in der Simulation möglich, sodass die Güte des Moduls gemessen und bewertet werden kann.

Der Testbaukasten wurde bisher nicht auf makroskopischer Ebene eingesetzt. Dementsprechend muss die Anwendbarkeit des Testbaukastens auf dieser Ebene noch gezeigt werden.

### 9.4.8 Führung

Für einen Testtreiber des Moduls der Führung wird eine subjektive Szene und eine straßengenaue Route benötigt. Die Ausgangsschnittstelle ist eine Sollpose mit Solldynamik. Für die Sollposenberechnung wird auf Basis der Ziele und Werte der automatisierten Fahrfunktion eine Selektion und Augmentation der subjektiven Szene vorgenommen, um daraus eine Situation zu erstellen (Ulbrich u. a., 2015). Diese wird mittels der Situationsanalyse bewertet. Darauf aufbauend werden Fahrmanöver in der Verhaltensplanung ausgewählt, um die Mission in der Situation zu erfüllen. Aus den Fahrmanövern werden Sollposen mit Solldynamiken bestimmt. Diese werden der Stabilisierungsebene übergeben.

Die straßengenaue Route und die objektive Szene werden im Testbaukasten aus der Simulation übernommen. Aus der objektiven Szene wird mit Hilfe des Frameworks von Lackmann (2015) eine subjektive Szene erstellt. Damit ist der Testbaukasten in der Lage, die Eingangsdaten für das Modul der Führung bereitzustellen. Die Blöcke innerhalb des Moduls der Führung sind jedoch nur bedingt mit Hilfe der Simulation testbar. Dies ist damit begründet, dass für die Blöcke nur unter Randbedingungen die Eingangs- und Ausgangsdaten zur Verfügung gestellt werden können. Für die Erstellung von genauen Situationen wird beispielsweise ein genaues Modell der Szenenselektion und -augmentation benötigt, um die genaue Zielsituation für die jeweilige Szene zu erhalten. Die Situationen und die Verhaltensentscheidungen sind aber funktionsabhängig, sodass womöglich die gewünschten Ausgangsdaten des Moduls für die jeweilige subjektive Szene nicht genau beschrieben werden können. Damit besteht nur bedingt die Möglichkeit, Modelle für das Modul der Führung für die Simulation zu erstellen. Diese Aussage trifft auch Ulbrich (2016, S.137), der die Bereitstellung von genauen Daten für die Module der Führung als einen Forschungsgegenstand beschreibt. Trotz dieser Herausforderung kann die Simulation unter bestimmten Randbedingungen für den Test des Moduls der Führung eingesetzt werden. Für diesen Schritt ist eine Abstraktion der automatisierten Fahrfunktion notwendig, die beispielsweise aus der Spezifikation abgeleitet wird. Mit dieser abstrahierten Fahrfunktion wird getestet, ob sich die zu testende Fahrfunktion gemäß den Spezifikationen korrekt verhält. Ein möglicher Einsatz einer solchen Abstraktion wird von Mauritz u. a. (2015) vorgestellt. Mit Hilfe der Abstraktion wird die Möglichkeit geschaffen, die grundsätzliche Funktionalität der Situationsanalyse und der Verhaltensplanung mittels der Simulation zu testen. Ein Beispiel wäre die Überprüfung, ob die Entscheidung für einen Fahrstreifenwechsel für die Erfüllung einer Mission grundsätzlich korrekt oder falsch ist. Die Güte der einzelnen Module kann

jedoch nicht bestimmt werden, da die genauen Ausgangsdaten, aufgrund von fehlenden Simulationsmodellen für eine perfekte Fahrfunktion, nicht erstellbar sind.

Die Erstellung von abstrakten Fahrfunktionen auf Basis der Spezifikation ist derzeit im Testbaukasten nicht realisiert. Zurzeit wird die Szene als Eingangsgröße für das Modul der Führung zur Verfügung gestellt. Dadurch besteht die Möglichkeit, das Modul als Ganzes zu testen. Die Ausgangsdaten in Form von Sollposen sind mit den Ausgängen von Fahrermodellen, die in der Simulation verfügbar sind, vergleichbar. Die Güte des Moduls ist jedoch nicht bewertbar, da keine genauen Ausgangsdaten aus der Simulation zum Vergleich zur Verfügung stehen.

### 9.4.9 Stabilisierung

Für den Testtreiber des Moduls der Stabilisierung wird zum einen die Sollpose mit der Solldynamik benötigt. Zum anderen sind die identifizierten Merkmale der Merkmalsextraktion und modellbasierten Filterung erforderlich. Innerhalb des Moduls wird auf Basis der Sollpose inklusive Solldynamik eine Trajektorienplanung durchgeführt. Mittels der Regelung werden die Sollgrößen für die Aktorik berechnet, die über die Ausgangsschnittstelle an die Aktorik übergeben werden.

Die Sollposen inklusive der Solldynamik können aus der Simulation erstellt werden. Hierzu werden Fahrermodelle verwendet, welche die Sollpose für den nächsten Simulationsschritt vorgeben. Eine Übernahme der Sollpose aus der Führung ist nur mittels konkreter Fahrfunktion möglich. Die extrahierten Merkmale aus der Merkmalsextraktion und der modellbasierten Filterung können ebenfalls aus der Simulation mit Hilfe von Sensormodellen erstellt werden. Damit besteht die Möglichkeit, alle Eingangsdaten für den Block der Stabilisierung zur Verfügung zu stellen.

Die Simulation bietet derzeit keine Möglichkeit, genaue Ausgangsdaten für das Modul der Stabilisierung zu erstellen, wodurch keine Gütebestimmung der Stabilisierung mit Hilfe der Simulation möglich ist. Mögliche Ausgangsgrößen können in der Simulation durch Fahrermodelle berechnet werden, die über eine eigene Trajektorienplanung und Regelung verfügen. Mit diesen kann getestet werden, ob das grundsätzliche Verhalten der Trajektorienplanung dem Wunschverhalten entspricht oder wie „gut“ das Testobjekt gegenüber dem Fahrermodell die Sollpose und Solldynamik erreicht. Hierfür werden von der Simulation unabhängige Bewertungskriterien benötigt, wie sie in Kapitel 10 vorgestellt werden.

Derzeit beinhaltet der Testbaukasten keine eigenen Fahrermodelle. Jedoch besteht die Möglichkeit, die Fahrermodelle, die in verschiedenen Simulationsumgebungen verfügbar sind, zu nutzen und daraus die Sollpose inklusive der Solldynamik zur Verfügung zu stellen.

### 9.4.10 Aktorik

Für einen Testtreiber der Aktorik werden die Sollgrößen für die verschiedenen Aktorikkomponenten benötigt. Diese können in der Simulation mit Hilfe von Fahrermodellen erstellt werden. Mit diesen Sollgrößen besteht die Möglichkeit, reale Aktorikkomponenten oder Aktorikmodelle zu testen. Dabei kann beispielsweise die Reaktionszeit überprüft werden. Ebenfalls kann das grundsätzliche Verhalten der Simulationsmodelle getestet werden. Für eine Bewertung der Modelle werden Vergleichsdaten benötigt, die beispielsweise in Versuchen mit realen Aktoren bestimmt werden.

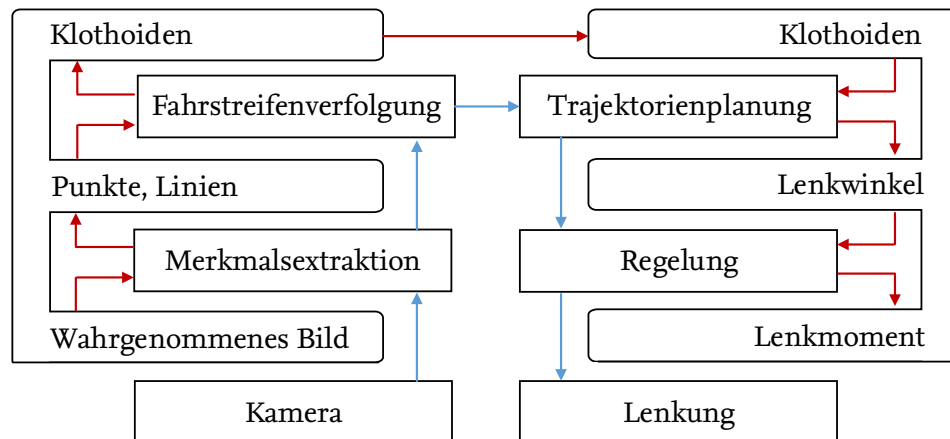
Der Testbaukasten verfügt derzeit über keine Aktorikmodelle. Bei den bisherigen Anwendungen war die Aktorik in der Simulation ein Teil des Fahrdynamikmodells oder eine reale



Komponente im Versuchsträger. Ein Test der eigentlichen Aktorik wurde mit dem Testbaukasten bisher nicht durchgeführt. Der Nachweis der Anwendbarkeit muss dementsprechend noch erbracht werden.

#### 9.4.11 Exemplarische Anwendung der Testtreiber an einer Querführungsfunktion

Im Folgenden wird die Anwendung von verschiedenen Testtreibern exemplarisch anhand der Fahrfunktion erläutert, welche in Kapitel 3.3 vorgestellt wurde. Abbildung 9.7 zeigt den schematischen Aufbau der Fahrfunktion mit möglichen Schnittstellen anhand der Module der funktionalen Systemarchitektur aus Kapitel 3.2. Die blauen Pfeile zeichnen den Weg der Informationen ohne Einbindung der Testtreiber. Die roten Pfeile zeigen die Möglichkeiten der Integration von Testtreibern an verschiedenen Stellen der Fahrfunktion.



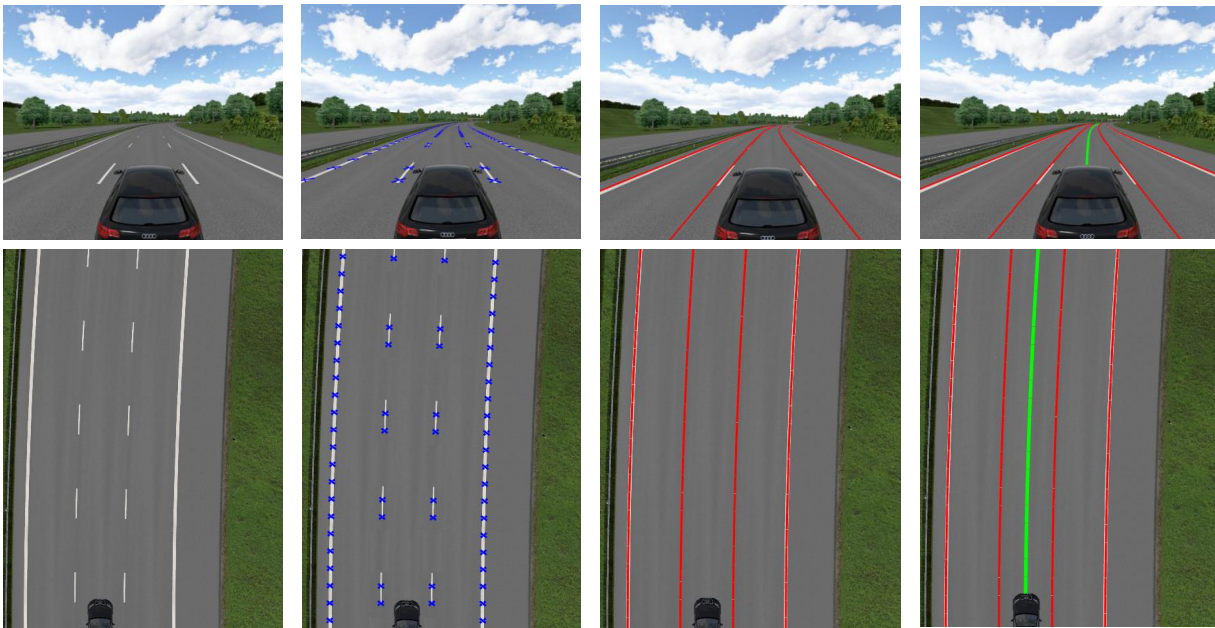
**Abbildung 9.7:** Testtreiber für die exemplarische Fahrfunktion aus Kapitel 3.3. Die blauen Pfeile zeigen den Weg der Informationen ohne Einbindung der Testtreiber. Die roten Pfeile zeigen die Möglichkeiten die Testtreiber zu integrieren.

Für die Realisierung der Fahrfunktion werden Bilder einer Szene durch eine Kamera wahrgenommen. Mittels der Merkmalsextraktion werden aus den Bildern Merkmale, wie Punkte oder Linien, extrahiert, die Bestandteile eines Fahrstreifens sein könnten. Die Merkmale werden der Fahrstreifenverfolgung übergeben, die auf dieser Basis den Verlauf der Fahrstreifen schätzt und beispielsweise in Form von Klothoiden an die Trajektorienplanung übergibt. Die Trajektorienplanung berechnet die Trajektorie, um dem Fahrstreifen zu folgen und übergibt der Regelung den dafür notwendigen Lenkwinkel. Von der Regelung werden die benötigten Stellgrößen berechnet, um den vorgegebenen Lenkwinkel zu erreichen. Für das Beispiel der Querführungsfunktion könnte dies das Lenkmoment sein, welches an die Lenkung übergeben wird.

Durch die in Abbildung 9.7 dargestellten Testtreiber wird die Möglichkeit geschaffen, die Querführungsfunktion als Ganzes zu testen. Dafür werden zum einen synthetische Bilder einer Szene benötigt. Zum anderen ist ein Lenkungsmodell erforderlich, welches ein Lenkmoment verarbeiten kann. Weiterhin besteht die Möglichkeit, einzelne Teile der Fahrfunktion zu testen. So können beispielsweise die Punkte, Linien oder Klothoiden der Fahrstreifen aus der Simulation mit den von der Merkmalsextraktion und Fahrstreifenverfolgung erzeugten Daten



verglichen werden. Damit können die Merkmalsextraktion und die Fahrstreifenverfolgung mit Hilfe der Simulation sowohl getestet als auch bewertet werden. Die Module der Trajektorienplanung und Regelung können, wie in den vorherigen Kapiteln erläutert, lediglich getestet werden. Für eine Bewertung fehlt die Möglichkeit, die genauen Daten aus der Simulation zu generieren. Abbildung 9.8 zeigt beispielhaft die Erzeugung von synthetischen Bildern und die daraus erstellten genauen Informationen über den Fahrstreifenverlauf in Form von Punkten, Linien und einer Klothoide des eigenen Fahrstreifens.



**Abbildung 9.8:** Erstellung von Testdaten aus der Simulation in Form von Rohbildern, Punkten (blau), Linien (rot) und einer Klothoide (grün) des eigenen Fahrstreifens. Oben: Perspektive des Fahrzeugs. Unten: Draufsicht.<sup>21</sup>

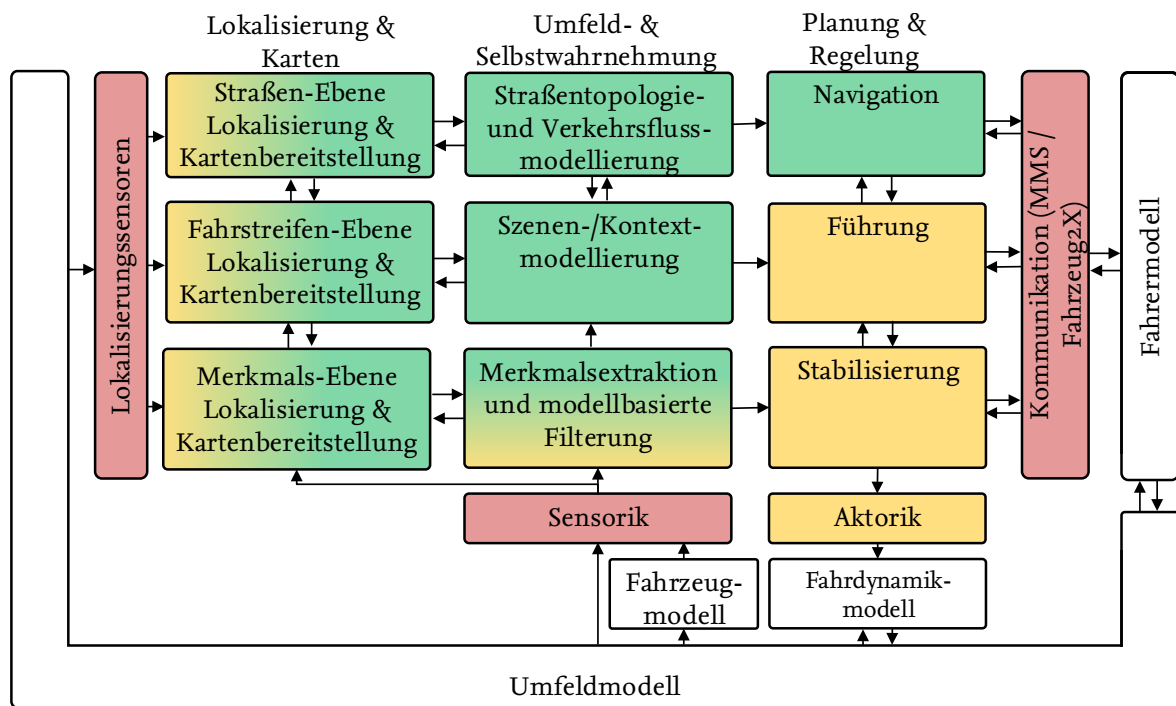
### 9.4.12 Zusammenfassung

Dieses Unterkapitel hat mögliche Testtreiber für den Test von Modulen der funktionalen Systemarchitektur beschrieben. In diesem Rahmen wurde gezeigt, welche Modelle in der Simulation für den Test der Module benötigt werden und inwiefern der Testbaukasten in der aktuellen Konzeption für den Test der verschiedenen Module verwendbar ist. Anhand der Fahrfunktion aus Kapitel 3.3 wurde die Anwendung von verschiedenen Testtreibern exemplarisch erläutert. Abbildung 9.9 stellt das Ergebnis der Analyse und die Bewertung bezüglich der Testbarkeit der Module der funktionalen Systemarchitektur mittels des Testbaukastens dar.

Die rot eingefärbten Module der Architektur lassen sich aufgrund von fehlenden Metriken und zum Teil fehlenden Modellen sowie notwendigen Emittlern derzeit nicht mit dem Testbaukasten testen. Dies betrifft die Module der Sensorik, Lokalisierungssensoren und Kommunikation.

Die gelben Blöcke sind testbar, aber nicht mit Hilfe der Simulation bewertbar. Die grundsätzliche Funktionalität der Blöcke kann getestet werden, da die Eingangsdaten

<sup>21</sup>Die Bilder wurden mit der Simulationssoftware *Virtual Test Drive* der Firma *Vires* erstellt.



**Abbildung 9.9:** Erweiterung der Architektur nach Matthaei (2015) um die benötigten Modelle für den Test mittels Simulation. Legende: grüne Einfärbung: Modul kann getestet und bewertet werden, gelbe Einfärbung: Modul kann eingeschränkt getestet werden, rote Einfärbung: Modul kann nicht getestet werden

zur Verfügung gestellt werden können. Für eine Bewertung werden von der Simulation unabhängige Bewertungskriterien benötigt. In der Spalte der Planung und Regelung können für die Module der Führung, Stabilisierung und Aktorik keine genauen Ausgangsdaten erstellt werden. Damit ist für diese Module keine Bewertung auf Basis von genauen Daten möglich. Für das Modul der Merkmalsextraktion und modellbasierten Filterung sowie für die Spalte der Lokalisierung und Karten können die genauen Ausgangsdaten erstellt werden. Jedoch liegt die Herausforderung in der Erstellung von Eingangsdaten, die von der Umfeldsensorik und den Lokalisierungssensoren übertragen werden. Für diese Daten fehlen derzeit Metriken, um die Güte der Modelldaten zu bewerten.

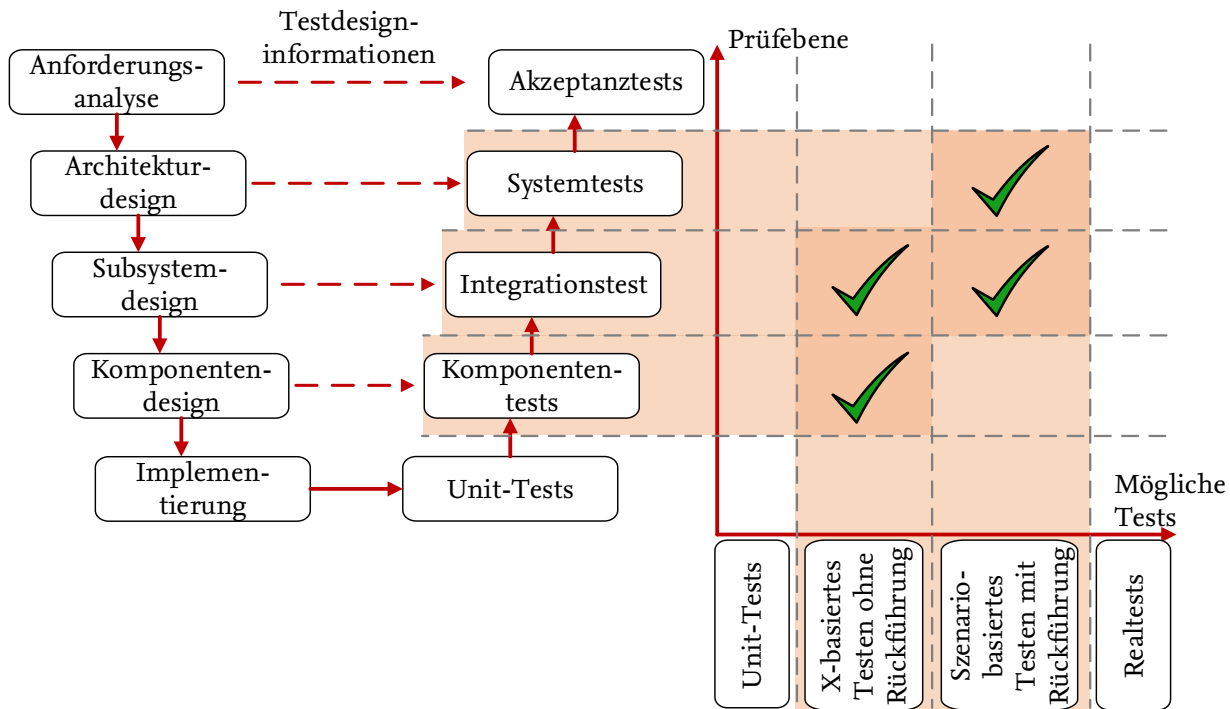
Die grün eingefärbten Blöcke können mit dem Testbaukasten sowohl getestet als auch bewertet werden, da sowohl die genauen Eingangs- als auch Ausgangsdaten aus der Simulation erzeugt werden können und auf dieser Basis eine Bewertung der Module durchführbar ist.

## 9.5 Zusammenfassung

Dieses Kapitel hat die Testdurchführung des modularen virtuellen Testbaukastens vorgestellt. Dafür wurden zunächst Anforderungen an die Testdurchführung identifiziert. Darauf aufbauend wurden die Möglichkeiten der Testdurchführung im öffentlichen Straßenverkehr, auf dem Testgelände und mittels X-in-the-Loop-Verfahren anhand der Anforderungen bewertet. Auf Realtests im öffentlichen Straßenverkehr oder auf dem Testgelände kann aufgrund von fehlenden Metriken für die Bewertung von Simulationsmodellen weiterhin nicht verzichtet werden. Die X-in-the-Loop-Verfahren können als ergänzende Testwerkzeuge

eingesetzt werden. Für die Zuordnung von Testfällen auf die X-in-the-Loop-Verfahren wurde eine zweistufige Methode vorgestellt, die geeignete Verfahren für die Testdurchführung identifiziert.

Weiterhin wurde anhand der funktionalen Systemarchitektur gezeigt, welche Module mit den aktuellen Ansätzen des modularen virtuellen Testbaukastens testbar sind. Mit den vorgestellten Testtreibern sind Komponenten- und Integrationstests in Form von X-basierten Tests ohne Rückführung möglich. Das X steht dabei stellvertretend für die Schnittstellen zwischen den Modulen. Weiterhin können Integrations- und Systemtests als Szenario-basierte Tests mit Rückführung durchgeführt werden. Abbildung 9.10 zeigt das V-Modell mit den vorgesehenen Prüfebenen erweitert um die Testebenen, die im modularen virtuellen Testbaukasten vorgesehen sind.



**Abbildung 9.10:** Integration des modularen virtuellen Testbaukastens in das V-Modell nach Rook (1986)

## 10 Testfallauswertung

Dieses Kapitel beschreibt die Testfallauswertung des modularen virtuellen Testbaukastens. Die Aufgabe ist die Bewertung und Analyse der Testergebnisse anhand von Daten, die während der Testdurchführung gemessen und/oder beobachtet werden. In der Bewertung wird ein Vergleich der Testergebnisse mit den erwarteten Ergebnissen durchgeführt. Die Analyse interpretiert die Ergebnisse in Bezug auf die Szenarioparameter. In diesem Kapitel werden zunächst Anforderungen an die Testfallauswertung definiert. Darauf aufbauend werden ausgewählte Methoden aus dem Stand der Technik zur Testfallauswertung vorgestellt, die das Potential haben die Anforderungen zu erfüllen.

### 10.1 Anforderungen

Die gestellten Anforderungen werden unterteilt in die Anforderungen an die Bewertung und an die Analyse der Testergebnisse. Die Anforderungen wurden auf Basis der Erfahrungen des Autors im Projekt des Engstellenassistenten erhoben.

#### 10.1.1 Anforderungen an die Bewertung von Testergebnissen

**Für die Testfallauswertung müssen Bewertungskriterien auf mikroskopischer Ebene definierbar sein.**

Es müssen Bewertungskriterien auf mikroskopischer Ebene definierbar sein. Die Bewertung der Testergebnisse findet auf Datenebene statt. Dies bedeutet, dass überprüft wird, ob sich die gemessenen und beobachteten Werte in einem definierten Wertebereich befinden. Dabei sollen verschiedene Funktionen für die Bewertung der Ergebnisse nutzbar sein, um unterschiedlichste Kriterien zu bewerten. Sobald sich die Eingangswerte für ein Kriterium außerhalb des definierten Wertebereichs befinden, wird der Testfall bezüglich des Bewertungskriteriums als *nicht bestanden* bewertet. Neben der Bewertung, ob der Testfall *bestanden* oder *nicht bestanden* wurde, soll die Bewertung eine Abstufung der Testergebnisse ermöglichen, um so eine Aussage zu treffen, wie „gut“ ein Testfall bestanden wurde. Dabei wird gefordert, dass die Bewertungsfunktion eine Monotonie aufweist, sodass größere Abweichungen vom Zielwert durchgängig zu „schlechteren“ Testergebnissen führen.

**Für die Testfallauswertung sollen Bewertungskriterien auf mesoskopischer Ebene definierbar sein.**

Es sollen Bewertungskriterien auf mesoskopischer Ebene definierbar sein. Ein Beispiel für ein mesoskopisches Bewertungskriterium ist die Durchführung eines Fahrstreifenwechsels oder eines Fahrmanövers in einem definierten Streckenabschnitt. Ein nicht durchgeführter Fahrstreifenwechsel, der für die Erfüllung der Mission notwendig ist, führt dazu, dass der Testfall *nicht bestanden* wird. Die Bewertungen auf dieser Ebene sollen auf den Bewertungskriterien der mikroskopischen Ebene, d.h. auf Datenebene, aufbauen und/oder auf Informationen, die nur auf mesoskopischer Ebene verfügbar sind.

**Für die Testfallauswertung sollen Bewertungskriterien auf makroskopischer Ebene definierbar sein.**

Es sollen Bewertungskriterien auf makroskopischer Ebene definierbar sein. Die Bewertungskriterien sollen ihre Informationen aus der mesoskopischen und mikroskopischen Ebene erhalten. Die grundsätzliche Erfüllung der Mission des Fahrzeugs oder die Energie beziehungsweise Zeit, die für die Erfüllung der Mission benötigt wurden, sind Beispiele für makroskopische Bewertungskriterien.

**Die Testfallauswertung soll eine Vergleichbarkeit von Testergebnissen ermöglichen.**

Die Testfallauswertung soll die Möglichkeit schaffen, die Ergebnisse von verschiedenen Testfällen zu vergleichen. Dazu sollen Methoden angeboten werden, die eine Normierung der Bewertungskriterien ermöglichen.

### 10.1.2 Anforderungen an die Analyse von Testergebnissen

**Die Testfallauswertung soll Methoden anbieten, um Rückschlüsse auf die Ursache der Testergebnisse zu ziehen.**

Die Testfallauswertung soll die Testergebnisse analysieren, um Rückschlüsse auf die Ursachen von Testergebnissen zu ermöglichen. Dies bedeutet, dass eine Veränderung der Testergebnisse auf eine Veränderung der Parameter oder Parameterstufen zurückgeführt werden kann.

**Die Testfallauswertung soll die Identifikation von Haupt- und Wechselwirkungen der Einflussparameter ermöglichen.**

Die Testfallauswertung soll bei der Analyse der Testergebnisse Haupt- und Wechselwirkungen der Einflussparameter identifizieren. Die Hauptwirkung einer Parameterstufe ist dabei die gemittelte Auswirkung auf ein Bewertungskriterium. Weiterhin sollen sich verstärkende und abschwächende Wechselwirkungen identifiziert werden.

Diese Anforderung wird in der vorliegenden Arbeit nicht erfüllt. Trotzdem wird die Anforderung genannt, um zu zeigen, dass die Anforderung bei der Erstellung des Konzepts bereits bedacht wurde und auf diesem Gebiet weitere Forschungsarbeiten notwendig sind. Die Erfüllung der Anforderungen stellt besonders bei automatisierten Fahrfunktionen eine Herausforderung dar, da sich die Systeme, wie in der Einleitung beschrieben, in einer offenen Umgebung befinden, in der eine Vielzahl von Einflussparameter in Wechselwirkung treten.

Um einen Forschungsbeitrag zur Erfüllung dieser Anforderung zu leisten, wird in der vorliegenden Arbeit analysiert, ob sich Haupt- und Wechselwirkungen der Einflussparameter in der Fallstudie des Engstellenassistenten identifizieren lassen.

## 10.2 Methoden aus dem Stand der Technik

In diesem Unterkapitel werden relevante Methoden aus dem Stand der Technik zur Testfallauswertung vorgestellt. Die Testfallauswertung nutzt dabei Methoden der statistischen Versuchsplanung. Kacker u. a. (2013) zeigen, wie sich die Methoden der statistischen Versuchsplanung mit dem kombinatorischen Testen verbinden lassen, wenn in der erstellten Testsuite jede Parameterstufe gleich oft getestet wird.

Da die Methoden nur angewendet und nicht weiterentwickelt wurden, findet lediglich eine Vorstellung der Methoden statt. Ein Beitrag zum Stand der Forschung im Bereich der

Testfallauswertung wird vom Testbaukasten nicht beigetragen. Die Methoden werden jedoch erstmalig im Bereich des Testens von automatisierten Fahrfunktionen angewendet.

### 10.2.1 Qualitätsverlustfunktionen zur Bewertung von Testergebnissen

Als exemplarische Bewertungsfunktion nutzt der modulare virtuelle Testbaukasten derzeit die Qualitätsverlustfunktionen aus der Qualitätslehre von Taguchi u. a. (2005, Kap.8). Taguchi geht in seiner Qualitätslehre davon aus, dass immer eine Streuung bei der Messung der zu erwartenden Zielwerte vorhanden ist. Dieses wird durch die Einführung von Qualitätsverlustfunktionen beachtet. Die Funktionen beschreiben einen Qualitätsverlust  $L(y)$  für ein Bewertungskriterium, sobald ein Eingangswert  $y$  vom Zielwert  $m$  abweicht. Eingangswerte können unter anderem Messwerte oder Schätzwerte sein. Der maximal erlaubte Qualitätsverlust  $A_0$  wird bei  $m \pm \Delta_0$  erreicht. Die Qualitätsverlustfunktionen werden durch parabolische Funktionen modelliert. Durch die Einführung der Qualitätsverlustfunktionen werden die Testfälle nicht nur auf *bestanden* und *nicht bestanden* bewertet, sondern wie „gut“ ein Testfall bestanden wurde. Somit sollte der Qualitätsverlust  $L(y)$  möglichst klein sein, um ein „gutes“ Testergebnis zu erhalten. Die Qualitätsverlustfunktionen haben weiterhin den Vorteil, dass unterschiedliche Bewertungskriterien über den Qualitätsverlust normiert werden und damit vergleichbar sind. Durch die Wahl des maximalen Qualitätsverlusts werden die Bewertungskriterien relativ zueinander gewichtet.

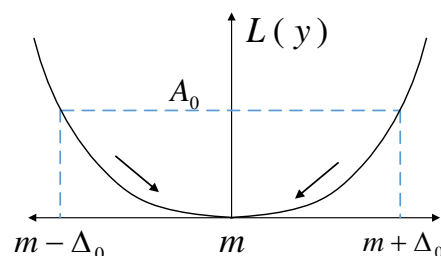
Nach Taguchi u. a. (2005, Kap.8) wird zwischen Zielwert-, Maximierungs- und Minimierungsfunktion sowie asymmetrischer Qualitätsverlustfunktion unterschieden. Die verschiedenen Qualitätsverlustfunktionen und mögliche Anwendungen werden im Folgenden erläutert.

#### Symmetrischer Zielwert

Die Qualitätsfunktion des symmetrischen Zielwerts bewertet die Messgröße  $y$  in Bezug zum Zielwert  $m$ . Dabei werden positive und negative Abweichungen gleich bewertet. Der Verlustkoeffizient  $k$  wird zur Normierung verwendet, um die Einheiten der Eingangsgrößen zu kürzen und so verschiedene Eingangsgrößen anhand des Qualitätsverlusts vergleichbar zu machen. Der Koeffizient bestimmt die Breite der Verlustparabel und berechnet sich aus der erlaubten Abweichung  $\Delta_0$  und dem maximalen Qualitätsverlust  $A_0$ .

In Abbildung 10.1 ist die Formel zur Berechnung des symmetrischen Zielwerts und die grafische Abbildung der Qualitätsverlustfunktion dargestellt. Ein mögliches Bewertungskriterium für einen symmetrischen Zielwert ist die Querbeschleunigung, die die Insassen während der Fahrt mit der in Kapitel 3.3 beschriebenen Fahrfunktion erfahren. Jede positive als auch negative Abweichung vom Zielwert führt zu einem Qualitätsverlust.

$$\begin{aligned} L(y) &= k(y - m)^2 \\ &= \frac{A_0}{\Delta_0^2} (y - m)^2 \end{aligned}$$

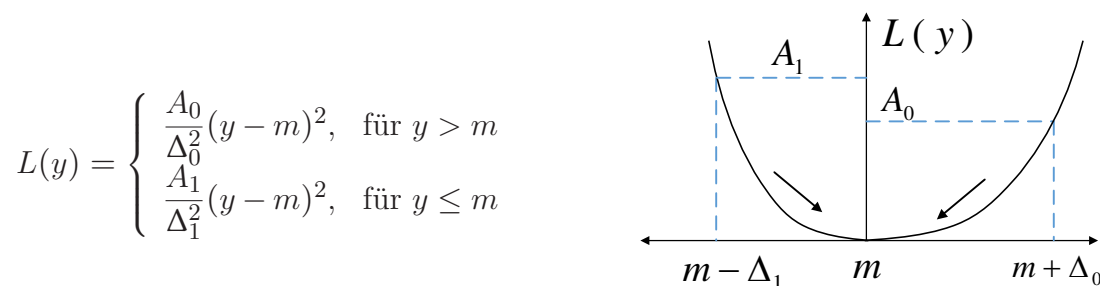


**Abbildung 10.1:** Berechnungsformel und grafische Darstellung der symmetrischen Qualitätsverlustfunktion



### Asymmetrischer Zielwert

Der asymmetrische Zielwert unterscheidet sich vom symmetrischen Zielwert dadurch, dass eine negative Abweichung anders als eine positive Abweichung gewichtet wird. Eine positive Abweichung wird durch den maximalen Qualitätsverlust  $A_0$  bei der Abweichung  $\Delta_0$  beschrieben. Die negative Abweichung wird durch die äquivalenten Variablen  $A_1$  und  $\Delta_1$  beschrieben. Abbildung 10.2 stellt die grafische Abbildung und die Formeln zur Berechnung des asymmetrischen Qualitätsverlusts dar. Ein Bewertungskriterium für die asymmetrische Qualitätsverlustfunktion könnte beispielsweise die Ablage des Fahrzeugs im Fahrstreifen sein. Wenn sich ein automatisiertes Fahrzeug mit der in Kapitel 3.3 beschriebenen Querführungsfunktion auf einer zweistreifigen Autobahn auf dem rechten Fahrstreifen befindet, könnte eine Abweichung nach links mit einem höheren Qualitätsverlust bewertet werden als eine Abweichung nach rechts. Dadurch wird die Gefahr bewertet, dass es auf der linken Seite wahrscheinlicher ist mit einem anderen Verkehrsteilnehmer zu kollidieren.



**Abbildung 10.2:** Berechnungsformel und grafische Darstellung der asymmetrischen Qualitätsverlustfunktion

### Minimierung

Kriterien, deren Eingangswerte idealerweise 0 sein sollen und Eingangswerte kleiner 0 nicht erreichen, können mit der Minimierungsfunktion bewertet werden. Diese ergibt sich aus der Formel für die symmetrische Zielfunktion und einem Zielwert  $m = 0$ . In Abbildung 10.3 sind die grafische Abbildung und die Formeln zur Berechnung des Qualitätsverlusts angegeben. Ein Beispiel für das Bewertungskriterium einer Minimierung stellt der Energieverbrauch des Fahrzeugs während der Fahrt dar. Dieser sollte möglichst gering sein. Die maximal erlaubte Abweichung ist bei diesem Beispiel direkt durch die maximal verfügbare Energie festgelegt.

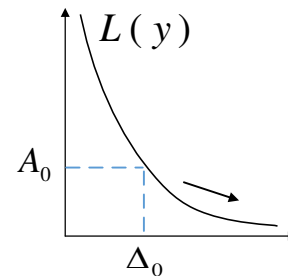


**Abbildung 10.3:** Berechnungsformel und grafische Darstellung der Qualitätsverlustfunktion zur Minimierung

### Maximierung

Für Bewertungskriterien, deren Eingangswerte zu maximieren sind, ist die Qualitätsverlustfunktion der Maximierung vorgesehen. Für diese ergibt sich ein theoretisch  $\infty$ -großer Zielwert. Eine Besonderheit ergibt sich für den Wert  $\Delta_0$ , welcher nicht die erlaubte Abweichung vom Zielwert definiert, sondern die erlaubte Abweichung zum Nullpunkt darstellt und somit den kleinsten erlaubten Wert beschreibt. In Abbildung 10.4 sind die grafische Abbildung und die Formeln zur Berechnung des Qualitätsverlusts angegeben. Ein Beispielkriterium für eine Maximierungsfunktion ist der Abstand zu stationären oder beweglichen Elementen während der Testdurchführungen. Dieser sollte während der Fahrt immer größer 0 sein, um Kollisionen zu vermeiden.

$$\begin{aligned} L(y) &= \frac{k}{y^2} \\ &= \frac{A_0 \Delta_0^2}{y^2} \end{aligned}$$



**Abbildung 10.4:** Berechnungsformel und grafische Darstellung der Qualitätsverlustfunktion zur Maximierung

Die gestellten Anforderungen an die Bewertung von verschiedenen Kriterien auf makroskopischer Ebene werden durch die Einführung der Qualitätsverlustfunktionen erfüllt. Die Funktionen besitzen die geforderte Monotonie bezüglich des Qualitätsverlusts. Eine Erhöhung der Abweichung vom Zielwert führt bei jeder Qualitätsverlustfunktion durchgängig zu einem erhöhten Qualitätsverlust. Weiterhin wird die Möglichkeit geschaffen, Bewertungskriterien auf mikroskopischer Ebene zu definieren und durch die Normierung auf den Qualitätsverlust zu vergleichen.

Die Qualitätsverlustfunktionen bieten jedoch nicht die Möglichkeit, Verknüpfungen von Bewertungskriterien auf verschiedenen Ebenen der funktionalen Systemarchitektur zu betrachten. Für die Verknüpfung von Qualitätsverlusten auf verschiedenen Ebenen der Architektur wäre eine ähnliche Herangehensweise zu betrachten wie Reschka (2016a, Kap.8) für die Fähigkeiten und Fertigkeiten von automatisierten Fahrzeugen vorschlägt. Mit einer baumähnlichen Struktur können die Qualitätsverluste der mikroskopischen Ebene zu Qualitätsverlusten auf mesoskopischer und makroskopischer Ebene fusioniert werden. Dies ist allerdings derzeit nur eine Idee, deren Realisierbarkeit weder in einem Konzept noch in einer Implementierung gezeigt wurde. Damit werden die Anforderungen nach Bewertungskriterien auf mesoskopischer und makroskopischer Ebene derzeit vom Testbaukasten nicht erfüllt.

### 10.2.2 Mittelwertanalyse zur Analyse von Testergebnissen

Der modulare virtuelle Testbaukasten nutzt zur Analyse der Testergebnisse die Mittelwertanalyse. Die Analyse wird für die qualitative Abschätzung der Parameterwirkungen der unterschiedlichen Diskretisierungsstufen auf die Bewertungskriterien verwendet. Die Wirkungen der Diskretisierungsstufen werden zwischen Haupt- und Wechselwirkungen unterschieden (Klein, 2014, S.76). Eine Hauptwirkung beschreibt die geschätzte Änderung eines

Bewertungskriteriums durch die Veränderung der Diskretisierungsstufe eines Parameters (Siebertz u. a., 2010, S.12f). Die Wechselwirkungen werden zwischen sich verstärkenden und sich abschwächenden sowie nicht-vorhandenen Wechselwirkungen unterschieden.

Zur Bestimmung der Haupt- und Wechselwirkungen wird der Mittelwert  $\bar{Y}_i$  der Eingangswerte  $y_{ik}$  über die Tests gebildet, bei denen die Diskretisierungsstufe  $i$  getestet wird (Klein, 2014, S.38ff), dargestellt in Gleichung 10.1. Die Hauptwirkung einer Diskretisierungsstufe  $\bar{Y}_i$  wird durch die Abweichung zu den übrigen Mittelwerten geschätzt.  $I$  beschreibt die Anzahl der Diskretisierungsstufen.  $n_i$  bezeichnet die Anzahl der Tests, die auf Diskretisierungsstufe  $i$  durchgeführt wurden.

$$\bar{Y}_i = \frac{1}{n_i} \sum_{k=1}^{n_i} y_{ik} \quad \forall \quad i = 1 \dots I \quad (10.1)$$

Weiterhin bietet die Mittelwertanalyse die Möglichkeit der Überprüfung, ob ein Mittelwert signifikant vom Gesamtmittelwert abweicht. Dazu wird basierend auf den Eingangswerten  $y_{ik}$  und einem Signifikanzniveau  $\alpha$  eine obere und untere Entscheidungsgrenze ( $UDL, LDL$ )<sup>1</sup> berechnet, um zu entscheiden, ob ein Mittelwert  $\bar{Y}_i$  signifikant vom Gesamtmittelwert  $\bar{\bar{Y}}$  abweicht. Dafür wird zunächst der Gesamtmittelwert  $\bar{\bar{Y}}$  über alle Mittelwerte  $\bar{Y}_i$  der Diskretisierungsstufen gebildet, dargestellt in Gleichung 10.2.

$$\bar{\bar{Y}} = \frac{1}{I} \sum_{i=1}^I \bar{Y}_i \quad (10.2)$$

Die obere und untere Entscheidungsgrenze berechnet sich aus der Gleichung 10.3 (Nelson u. a., 2005, S.35).  $m(\alpha; I; N - I)$  beschreibt einen multiplikativen Faktor, der abhängig vom Signifikanzniveau  $\alpha$ , von der Anzahl der Diskretisierungsstufen  $I$  und von der Anzahl der Tests  $N$  ist und die obere und untere Entscheidungsgrenze verschiebt. Der Faktor ist beispielsweise in Nelson u. a. (2005, S.210ff) nachzuschlagen.

$$UDL_i, LDL_i = \bar{\bar{Y}} \pm m(\alpha; I, N - I) \sqrt{MS_e \frac{N - n_i}{N n_i}} \quad (10.3)$$

$$MS_e = \frac{(n_1 - 1)s_1^2 + (n_2 - 1)s_2^2 + \dots (n_I - 1)s_I^2}{N - I} \quad (10.4)$$

$$s_i^2 = \frac{1}{n_i - 1} \sum_{k=1}^{n_i} (\bar{Y}_i - y_{ik})^2 \quad (10.5)$$

$$N = \sum_{i=1}^I n_i \quad (10.6)$$

Die Wechselwirkungen werden ebenfalls über die Mittelwerte geschätzt. Dazu werden jeweils zwei Parameterstufen gleichzeitig verglichen. Eine Möglichkeit der grafischen Darstellung von Wechselwirkungen bietet die *Response Surface* (Box und Draper, 2007, S.2). Bei dieser

<sup>1</sup>UDL: Upper Decision Lane, Frei übersetzt: obere Entscheidungslinie, LDL: Lower Decision Lane, Frei übersetzt: untere Entscheidungslinie.

werden auf der x- und y-Achse jeweils die Diskretisierungsstufen aufgetragen. Auf der z-Achse wird der Eingangswert dargestellt, der auf den jeweiligen Diskretisierungsstufen gemessen wurde. Für die Testfallauswertung im Testbaukasten wird der mittlere Qualitätsverlust aufgetragen, um die Wechselwirkungen zu visualisieren.

Die Mittelwertanalyse bietet das Potential, die gestellten Anforderungen an die Analyse der Testergebnisse zu erfüllen. Dies muss jedoch in einer Anwendung noch nachgewiesen werden. In der vorliegenden Arbeit wird die Mittelwertanalyse in der Fallstudie des Engstellenassistenten angewendet, die in Kapitel 12 vorgestellt wird. Damit wird evaluiert, ob die Mittelwertanalyse für die Analyse der Testergebnisse verwendbar ist.

### 10.3 Zusammenfassung

In diesem Kapitel wurden Methoden zur Bewertung und Analyse von Testergebnissen aus dem Bereich der statistischen Versuchsplanung vorgestellt. Diese Methoden werden im modularen virtuellen Testbaukasten für die Testfallauswertung verwendet und in der Fallstudie des Engstellenassistenten eingesetzt. Durch den Einsatz der Qualitätsverlustfunktionen wurde die Möglichkeit geschaffen, verschiedenste Bewertungskriterien auf mikroskopischer Ebene zu normieren. Damit sind die verschiedenen Bewertungskriterien vergleichbar und können in Bezug zueinander bewertet werden. Die Analyse der Mittelwerte bietet das Potential, die Haupt- und Wechselwirkungen von Einflussparametern zu schätzen. Im Bereich der statistischen Versuchsplanung stehen noch viele weitere Methoden zur Bewertung und Analyse von Testergebnissen zur Verfügung. Diese wurden in der vorliegenden Arbeit nicht weiter betrachtet. Die Anwendbarkeit der Methoden im Bereich des Testens von automatisierten Fahrfunktionen sollte jedoch weiter untersucht werden, um die Methoden gegebenenfalls in Folgearbeiten in den Testbaukasten zu integrieren.



# 11 Limitationen des modularen virtuellen Testbaukastens

In den vergangenen Kapiteln wurden der Aufbau sowie die einzelnen Module des modularen virtuellen Testbaukastens vorgestellt. Dieses Kapitel beschreibt darauf aufbauend die Limitationen des modularen virtuellen Testbaukastens. Viele Module des Testbaukastens sind derzeit nur als Konzepte vorhanden. Die Module, die umgesetzt wurden, sind im Projekt des Engstellenassistenten prototypisch entstanden und werden in Kapitel 12 vorgestellt. Eine Anwendung auf automatisierte Fahrfunktionen mit weiteren Automatisierungsgraden wurde bisher nur in einem eingeschränkten Rahmen exemplarisch anhand der in Kapitel 3.3 vorgestellten Fahrfunktion gezeigt. Dabei wurde gezeigt, dass die Anwendung grundsätzlich möglich ist. Ein praktischer Nachweis der Übertragbarkeit des Testkonzepts auf andere Automatisierungsgrade muss noch erbracht werden. Zur Beschreibung der Limitationen wird die Erfüllung der gestellten Anforderungen an ein Testkonzept aus Kapitel 5 bewertet. Weiterhin werden die Limitationen der einzelnen Module des Testbaukastens beschrieben.

## 11.1 Erfüllung der gestellten Anforderungen an ein Testkonzept

### 11.1.1 Effektivität

**Der Testbaukasten muss repräsentative und valide Testfälle für das Testobjekt ableiten.**

Durch das Modul der Auswahl und Analyse von Einflussparametern bietet der Testbaukasten die Möglichkeit, aus verschiedenen Quellen Informationen über die Umgebung des Testobjekts und damit über die Einflussparameter zu erhalten. Damit hat der Testbaukasten das Potential, systematisch repräsentative Testfälle abzuleiten. Der Testbaukasten bietet jedoch keine Möglichkeit zu zeigen, wann die benötigte Testabdeckung für die Freigabe des Systems durch repräsentative Testfälle erreicht wurde. Damit wird diese Anforderung vom Testbaukasten nur bedingt erfüllt, da die Einflussparameter durch das vorgestellte Schema systematisch beschreibbar sind, jedoch die Vollständigkeit der Testfälle für eine Freigabe des Systems nicht sichergestellt wird. Die Erfüllung dieser Anforderung wird im Forschungsprojekt *PEGASUS*<sup>1</sup> (Plättner u. a., 2016) weiterverfolgt.

**Der Testbaukasten muss eine variierbare Testfallerstellung und Testdurchführung haben.**

Durch den Einsatz der Simulation in Verbindung mit verschiedenen X-in-the-Loop-Verfahren werden Verfahren zur Testdurchführung gewählt, die die Umgebung des Testobjekts simulieren und damit eine hohe Variabilität der Umgebung ermöglichen. Weiterhin besteht die

---

<sup>1</sup>PEGASUS: Projekt zur Etablierung von generell akzeptierten Gütekriterien, Werkzeugen und Methoden sowie Szenarien und Situationen zur Freigabe hochautomatisierter Fahrfunktionen



Möglichkeit, durch das 4-Ebenen-Modell Szenarien zu definieren, die eine hohe Variabilität aufweisen. Damit wird diese Anforderung vom Testbaukasten erfüllt.

**Der Testbaukasten muss Messgrößen für die Testfallauswertung beobachtbar machen.**

Der Testbaukasten bietet derzeit keine zusätzlichen Möglichkeiten, um nicht messbare Werte zu beobachten. Aktuell werden nur die Messgrößen erfasst, die direkt von der Simulation oder vom Testwerkzeug zur Verfügung gestellt werden. Regelungstechnische Werkzeuge, die dazu dienen, nicht messbare Größe zu erfassen, wie ein Beobachter, sind derzeit im Testbaukasten nicht vorhanden. Damit wird diese Anforderung nur bedingt erfüllt, da die Beobachtbarkeit von Messgrößen vom gewählten Testwerkzeug abhängt.

### 11.1.2 Effizienz

**Das Testkonzept muss eine ökonomische Testdurchführung ermöglichen.**

Durch die vorgestellte Zuordnungsmethode unterstützt der Testbaukasten bei der Auswahl von Testwerkzeugen zur Testdurchführung. Die Bewertungskriterien der Zeit und Kosten für den Einsatz eines Testwerkzeugs können bereits durch eine gewichtete Bewertungsfunktion in der Zuordnungsmethode beachtet werden. Ob diese Auswahl auch in Bezug zur Ergebnisgüte ökonomisch ist, kann derzeit nicht beantwortet werden, da die Modellgüte aufgrund von fehlenden Metriken nicht in der Zuordnungsmethode beachtet wird. Daher kann die Erfüllung der Anforderung an dieser Stelle nicht überprüft werden.

**Der Testbaukasten muss reproduzierbar Testfälle durchführen können.**

Durch die vorgestellte Regelung der Akteure im 4-Ebenen-Modell besteht die Möglichkeit, mit Hilfe des Testbaukastens reproduzierbare Szenarien aus globaler Sicht als auch aus Sicht des Testobjekts zu erstellen. Symkenberg (2014) zeigt in ihrer Arbeit einen Nachweis für die Reproduzierbarkeit von Szenarien aus Sicht des Testobjekts. Die Anforderung der Reproduzierbarkeit wird durch das 4-Ebenen-Modell nur bei X-in-the-Loop-Verfahren erfüllt, bei denen die Akteure simuliert werden. Bei allen anderen Verfahren hat der Testbaukasten keine Möglichkeit einen Einfluss auf die Elemente zu nehmen. Damit wird die Anforderung unter der Bedingung von simulierten Akteuren vom Testbaukasten erfüllt.

**Der Testbaukasten muss frühzeitig im Entwicklungsprozess einsetzbar sein.**

Das Konzept des Testbaukastens ermöglicht die Durchführung von Komponententests, Integrationstests sowie Systemtests. Damit kann der Testbaukasten bezogen auf das V-Modell frühzeitig im Entwicklungsprozess für entwicklungsbegleitende Tests eingesetzt werden, wodurch die gestellte Anforderung erfüllt wird.

**Der Testbaukasten muss eine sichere Testdurchführung zur Verfügung stellen.**

Der Testbaukasten beachtet die Sicherheit des Testobjekts nur bei der Auswahl einer Testmethode. Bei der Bewertung der verschiedenen Testwerkzeuge wurde analysiert, dass bei allen Verfahren, bei denen die Szenerie oder die beweglichen Elemente emuliert werden oder real vorhanden sind, eine Gefahr für das Testobjekt besteht. Der Testbaukasten kann zur Erhöhung der Sicherheit während der Testdurchführung die entsprechenden Verfahren in der Zuordnungsmethode sperren beziehungsweise nicht beachten. Methoden, die das Testobjekt während der Testdurchführung schützen, sind derzeit im Testbaukasten nicht umgesetzt. Damit wird diese Anforderung vom Testbaukasten nur bedingt erfüllt.

### 11.1.3 Systematik

**Der Testbaukasten soll in vorhandene systematische Entwicklungsprozesse integrierbar sein.**

Der Testbaukasten lässt sich durch seinen Aufbau in den von Reschka u. a. (2015b, S.927) skizzierten systematischen Entwicklungsprozess integrieren. Durch den bereits vorhandenen Prozessschritt der *Validierung in Test und Simulation* besteht die Möglichkeit, den Testbaukasten an dieser Stelle zu integrieren, da dieser genau das Ziel der Validierung mit Hilfe der Simulation verfolgt. Jedoch wird der Prozessschritt, wie in Kapitel 5.1.3 beschrieben, erst durchgeführt, wenn die kleine Iteration erfolgreich durchgeführt wurde. Die Anwendung des Testbaukastens kann jedoch zeigen, dass die Fahrfunktion in der definierten Ausprägung nicht testbar ist, wodurch womöglich die Funktionsdefinition der Fahrfunktion geändert und anschließend eine Vielzahl von Entwicklungsschritten wiederholt werden muss. Daher wird an dieser Stelle eine Änderung des derzeitigen systematischen Entwicklungsprozesses vorgeschlagen. Wie in Kapitel 1.1 dargelegt, ist zur Realisierung einer Fahrfunktion nicht nur die technische Realisierung, sondern auch die Testbarkeit zu zeigen. Dadurch wird die Anwendung des Testbaukastens als vorgezogener Prozessschritt nach dem Schritt der *technischen Machbarkeit* vorgeschlagen. Damit wird die Möglichkeit geschaffen, nach der *technischen Machbarkeit* auch die *Testbarkeit* der Fahrfunktionen zu zeigen. Dies ist notwendig, um zukünftige automatisierte Fahrfunktionen bis zur Serienreife zu entwickeln. Sollte die Anwendung des Testbaukastens zeigen, dass die Fahrfunktion mit der gewählten Funktionsdefinition nicht testbar ist, kann in der kleinen Iterationsschleife die Definition der Fahrfunktion geändert werden und somit viele Ressourcen gespart werden.

**Der Testbaukasten muss eine systematische Erstellung, Durchführung und Auswertung von Testfällen ermöglichen.**

Der Testbaukasten ist durch seinen Aufbau als mehrstufiges Prozessmodell ein Konzept für die systematische Ableitung, Durchführung und Auswertung von Testfällen. Die Schnittstellen zwischen den Prozessschritten sind das vorgestellte Schema zur Beschreibung von Einflussparametern, die Beschreibung der logischen Szenarien im 4-Ebenen-Modell und die konkreten Testfälle. Die konkreten Testfälle werden durch logische Szenarien mit Szenarioparametern, Bewertungskriterien in Form von Qualitätsverlustfunktionen sowie den Anforderungen an die Testwerkzeuge in Form von Kiviat-Diagrammen beschrieben. Durch die Festlegung der Schnittstellen besteht das Potential, innerhalb der Module Methoden auszutauschen, sobald die Methoden die Schnittstellen bedienen. Dadurch ist es möglich, die Methoden an das jeweilige Testobjekt anzupassen, wodurch der Testbaukasten die Anforderung der systematischen Erstellung, Durchführung und Auswertung von Testfällen erfüllt.

## 11.2 Erfüllung der gestellten Anforderungen durch Anwender

**Der Testbaukasten muss standardisierte Beschreibungsformate verwenden.**

Der Testbaukasten nutzt derzeit zum Teil standardisierte Formate für die Schnittstellen zwischen den Prozessschritten. Für die Beschreibungen der Einflussparameter und Bewertungskriterien werden keine standardisierten Formate verwendet. Zur Beschreibung des

4-Ebenen-Modells wird das standardisierte *OpenDrive*-Format verwendet. Die Konzepte des 4-Ebenen-Modells sind in den Entwurf des neuen vorgeschlagenen Standards *OpenScenario* eingeflossen. Damit besteht das Potential, dass die Anforderungen der standardisierten Formate für die Beschreibung von Szenarien erfüllt werden. Diese Anforderung wird jedoch nicht als erfüllt angesehen, da sich das Format derzeit noch in der Weiterentwicklung befindet. Für die übrigen Module wird diese Anforderung derzeit nicht erfüllt.

**Der Testbaukasten muss eine manuell und automatisch anpassbare Testfallerstellung ermöglichen.**

Die Anwender des Testbaukastens haben derzeit die Möglichkeit, über menschenlesbare Schnittstellen die Parameter und Diskretisierungsstufen der Tests zu verändern. Damit ist eine manuelle Änderung der Parameter und Diskretisierungsstufen durch die Anwender möglich. Eine automatische Ableitung von neuen Testfällen aus vorausgegangenen Testergebnissen, wie beim evolutionären Testen, ist derzeit im Testbaukasten nicht möglich. Momentan ist nur eine kombinatorische Testfallableitung umgesetzt. Daher wird diese Anforderung nur bedingt erfüllt.

**Der Testbaukasten muss die Testergebnisse automatisch dokumentieren.**

Eine automatische Dokumentation der Testergebnisse ist derzeit im Testbaukasten nicht umgesetzt. Daher wird diese Anforderung vom Testbaukasten nicht erfüllt.

## 11.3 Erfüllung der gestellten Anforderungen durch Testobjekte

### Integrationsgrad des Testobjekts in der funktionalen Systemarchitektur

Die Möglichkeit, das Testobjekt abhängig von seinem Integrationsgrad in der funktionalen Systemarchitektur zu testen, wurde in Kapitel 9.4 diskutiert. Es wurde gezeigt, dass das Konzept des Testbaukastens die Möglichkeit bietet, die Module *Szenen-/Kontextmodellierung*, *Straßentopologie- und Verkehrsflussmodellierung* und *Navigation* der funktionalen Systemarchitektur zu testen und zu bewerten. Die Module *Merkmalsextraktion und modellbasierte Filterung*, *Lokalisierung & Karten*, *Führung*, *Stabilisierung* und *Aktorik* können mit Hilfe der Simulation getestet aber nicht bewertet werden. Die Module *Sensorik*, *Lokalisierungssensoren* und *Kommunikation* können derzeit nicht getestet werden. Daher wird diese Anforderung vom Testbaukasten nur bedingt erfüllt.

### Anforderungen durch verschiedene Automatisierungsgrade

In Kapitel 5.3.2 wurde dargestellt, dass die Anforderungen, die durch verschiedene Automatisierungsgrade an den Testbaukasten gestellt werden, zwischen den Graden 0 - 2 und 3 - 5 nach der Einteilung der SAE (2014) unterschieden werden.

Bei den Automatisierungsgraden 0 - 2 besteht die Anforderung an ein Fahrermodell, welches in der Lage ist, die Fahraufgabe zu übernehmen, um Übergabeszenarien zu testen. Weitere spezielle Anforderungen wurden nicht identifiziert. Da der Testbaukasten momentan keine Fahrermodelle zur Verfügung stellt, ist dieser auf die Modelle der Simulationsumgebungen oder auf ein Testwerkzeug mit realen Fahrern angewiesen. Wenn die Modelle der Simulationsumgebungen keine Übergaben zwischen automatisierter Fahrfunktion und Fahrermodell ermöglichen, kann der Testbaukasten diesen Test nicht durchführen.

Für die Automatisierungsgrade 3 - 5 wurde angenommen, dass das System viele der Fahrsituationen alleine lösen muss, da kein Fahrer als Rückfallebene zur Verfügung steht oder der Fahrer in der von Damböck (2013, S.148) definierten Zeitspanne von ca. 8 Sekunden die Fahrzeugführung nicht übernehmen kann. Bei einem Test der Systeme mit Automatisierungsgrad 3 - 4, bei denen der Fahrer potentiell die Fahrzeugführung übernehmen kann, werden Fahrermodelle benötigt, um die Übergabe der Fahrzeugführung, die sich in verschiedenen Szenarien gegebenenfalls als herausfordernd darstellt, zu testen. Für diese Tests gilt die schon beschriebene Erfüllung der Anforderungen. Bei den übrigen Tests, bei denen der Fahrer die Fahrzeugführung nicht übernehmen kann, und bei Systemen mit Automatisierungsgrad 5 wird kein Fahrermodell benötigt.

Für die Durchführung der Testfälle besteht die Herausforderung, dass die Systeme dauerhaft aktiv sind. Folglich können diese nicht deaktiviert werden, um beispielsweise den Start eines Testfalls in einer definierten Szene zu ermöglichen. Daher können die Systeme nicht ohne Hilfsmittel in die Zielszenen der Tests überführt werden, da die Systeme aktiv verhindern, die Zielszene zu erreichen. An dieser Stelle wird eine Simulation der Szenerie und der beweglichen Elemente benötigt, um Manöver auszuführen, die im Realtest nicht durchführbar sind, aber für das Erreichen der Zielszenen der Tests benötigt werden. Durch den Testbaukasten wird die Möglichkeit geschaffen, mit Hilfe des 4-Ebenen-Modells diese Manöver durchzuführen, wenn das Testwerkzeug die Szenerie und die beweglichen Elemente simuliert. Damit wird diese Anforderung mit den oben beschriebenen Einschränkungen vom Testbaukasten bedingt erfüllt.

## 11.4 Limitationen der Module des Testbaukastens

### Auswahl und Analyse

Bei der Analyse des Moduls Auswahl und Analyse von Einflussparametern wurde festgestellt, dass die Diskretisierung von wertkontinuierlichen Parametern eine Herausforderung für den Test automatisierter Fahrfunktionen darstellt und nach Kacker u. a. (2013) noch immer ein Forschungsthema ist. Nach der Analyse der Informationsquellen bietet derzeit keine Quelle die Möglichkeit, die Fehlerwahrscheinlichkeit aufgrund der Diskretisierung abzuschätzen. Der Testbaukasten kann derzeit nur ein Konzept zur Strukturierung der Einflussparameter zur Verfügung stellen. Eine Lösung für die Diskretisierung der Parameter stellt dies nicht dar, sodass dies immer noch eine offene Forschungsfrage darstellt.

In der Fallstudie des Engstellenassistenten wurden von den vorgestellten Informationsquellen primär die Normen und Richtlinien analysiert. Darauf aufbauend wurden Einflussparameter und Diskretisierungsstufen abgeleitet. Für dieses Assistenzsystem haben sich die Normen und Richtlinien als Informationsquelle besonders angeboten, da Arbeitsstellen Sonderszenarien darstellen, für die spezielle Normen und Richtlinien existieren. Zusätzlich konnte auf Verkehrszeichenpläne<sup>2</sup> von Firmen, die Arbeitsstellen planen und aufbauen, zurückgegriffen werden. Die Verkehrszeichenpläne passen die Arbeitsstellen basierend auf den Normen und Richtlinien an die jeweiligen Gegebenheiten der Autobahn an.

Für andere Assistenzsysteme oder automatisierte Fahrfunktionen, die für anderen Szenarien bestimmt sind, sind gegebenenfalls keine Normen und Richtlinien vorhanden. Die weiteren Nachteile der Normen und Richtlinien als Informationsquelle wurden bereits in Kapitel 7.6

<sup>2</sup>[https://mobil.hessen.de/irj/HSVV\\_Internet?cid=a6f243aaff7aedf0b2e4ff094c63e689](https://mobil.hessen.de/irj/HSVV_Internet?cid=a6f243aaff7aedf0b2e4ff094c63e689), abgerufen: 09.03.2016

zusammengefasst. Im Fall der nicht existierenden Normen und Richtlinien müssen andere Informationsquellen genutzt werden. Deren Anwendung im Testbaukasten wurde bisher nicht gezeigt.

### Testfallerstellung

Im Modul der Testfallerstellung wurde das 4-Ebenen-Modell zur Beschreibung von logischen Szenarien vorgestellt. Des Weiteren wurde eine systematische Testfallableitung vorgestellt, die eine Ableitung von logischen Szenarien zu konkreten Testfällen ermöglicht und dabei die Anzahl der Testfälle durch kombinatorische Algorithmen reduziert.

Durch das 4-Ebenen-Modell besteht die Möglichkeit, Szenarien strukturiert und systematisch aufzubauen. Da das Modell jedoch immer nur eine Vereinfachung der Realität darstellt, kann das Modell nie vollständig die Realität abbilden. Für den Engstellenassistenten wurden die Szenarien auf Basis der ersten drei Ebenen definiert. Die vierte Ebene konnte aufgrund der gewählten Testdurchführung nicht beeinflusst werden. Weiterhin wurden in der Fallstudie keine wetterabhängigen Modelle verwendet. Dadurch wurde ein Einsatz des 4-Ebenen-Modells mit Szenarien, die auf allen vier Ebenen spezifiziert wurden, bisher nicht gezeigt.

Für die Testableitung wurden Testmethoden verwendet, die das Testobjekt als Black-Box betrachten und auf Basis von Eingangs- und Ausgangsgrößen Testfälle ableiten. Methoden, die auf Basis des Quelltexts Testfälle ableiten, wurden nicht weiter betrachtet, da das Konzept des Testbaukastens davon ausgeht, dass der Quelltext des Testobjekts nicht zur Verfügung steht. Ein besonderer Fokus lag auf kombinatorischen Algorithmen zur Testfallableitung. Die Anwendung der Algorithmen wurde im Projekt des Engstellenassistenten für die Erstellung von stationären Szenarien, d.h. ohne sich bewegende Elemente, gezeigt. Die Anwendung für Szenarien mit beweglichen Elementen wurde bisher nicht gezeigt. Daher muss die Anwendbarkeit mit Szenarien inklusive beweglicher Elemente dementsprechend noch nachgewiesen werden.

### Testdurchführung

Die Testdurchführung des modularen virtuellen Testbaukastens wird mittels Simulationen und Emulationen realisiert. In Kapitel 9.2 wurde durch den Vergleich der gestellten Anforderung mit den Testdurchführungen im Realtest und in der Simulation gezeigt, dass derzeit auf Realtests für die Freigabe der Fahrfunktionen nicht verzichtet werden kann. Dies liegt vor allem an fehlenden Metriken für die Bewertung der Güte von Simulationsmodellen. Zusätzlich besteht die Herausforderung der Zertifizierung der Modelle und der Simulationssoftware, um einen Einsatz im Freigabeprozess zu erlauben. Aktuell gibt es zertifizierte Modelle, wie das Fahrdynamikmodell der Firma *IPG* im Pkw- als auch Lkw-Bereich (Baake u. a., 2014; Holzmann u. a., 2012; Wurster u. a., 2010). Eine komplett zertifizierte Simulationssoftware für den Test von automatisierten Fahrfunktionen ist nicht vorhanden, sodass die Simulation derzeit nicht für die Freigabe der Systeme verwendet werden darf (Bergmiller, 2016).

In der Simulation können derzeit Funktionstests durchgeführt werden, bei denen die grundsätzliche Funktionalität getestet wird. Für diese Tests kann die Simulation gewinnbringend eingesetzt werden, da eine Vielzahl von unterschiedlichsten Testfällen durchführbar ist. Für die Durchführung der Tests stehen verschiedene X-in-the-Loop-Verfahren zur Verfügung. Eine Methode für die Zuordnung von Testfällen auf die X-in-the-Loop-Verfahren wurde in Kapitel 9.3 vorgestellt. Mittels der Durchführung von Tests in der Simulation besteht

die Möglichkeit, die Anzahl von entwicklungsbegleitenden Realtests trotz der fehlenden Metriken zu reduzieren. Eine Vielzahl von vorbereitenden Tests, bei denen die grundsätzliche Funktionalität des Moduls getestet wird, können in der Simulation durchgeführt werden. Eine Optimierung oder Abstimmung der Module in der Simulation ist dagegen abhängig vom ausgewählten X-in-the-Loop-Verfahren nur bedingt möglich, sodass für diese Tests weiterhin Realtests notwendig sind.

### **Testfallauswertung**

Die Testfallauswertung des modularen Testbaukastens besteht aus der Bewertung der Testergebnisse mittels Qualitätsverlustfunktionen und der Analyse der Ergebnisse durch die Analyse der Mittelwerte. Weitere Verfahren sind derzeit nicht umgesetzt. Die Rückführung der Testergebnisse zur Auswahl und Analyse der Einflussparameter ist derzeit nur im Konzept vorgesehen und wurde bisher nicht umgesetzt beziehungsweise in einem Projekt gezeigt.





**Teil IV**

**Die Anwendung des modularen  
virtuellen Testbaukastens**



## 12 Der Engstellenassistent

Dieses Kapitel beschreibt die prototypische Umsetzung des modularen virtuellen Testbaukastens als Testkonzept für den Engstellenassistenten. Zunächst wird das Projekt des Engstellenassistenten und darauffolgend die realisierten Module der funktionalen Systemarchitektur aus Kapitel 3.2 vorgestellt. Im Anschluss wird in der Fallstudie der Testbaukasten als Testkonzept angewendet. Dazu werden Einflussparameter identifiziert, Testfälle abgeleitet, ein X-in-the-Loop-Verfahren für die Testdurchführung ausgewählt und die erstellten Testfälle durchgeführt. Anschließend werden die Testergebnisse anhand der vorgestellten Testfallauswertung bewertet und analysiert.

Teile dieses Kapitels wurden bereits in Schuldt u. a. (2014)<sup>1</sup> veröffentlicht.

### 12.1 Projektbeschreibung

Das folgende Kapitel beschreibt das Projekt des Engstellenassistenten. Dabei werden zunächst die Fahrfunktion und darauffolgend die realisierten Module der funktionalen Systemarchitektur vorgestellt.

#### 12.1.1 Funktionsbeschreibung

Der Engstellenassistent realisiert nach der Einteilung der SAE (2014) eine Fahrfunktion mit Automatisierungsgrad 1. Das System assistiert dem Fahrer bei der Querführung des Fahrzeugs und stellt eine Erweiterung des Fahrstreifenhalteassistenten dar. Die Funktionserweiterung besteht darin, dass nicht nur auf Fahrstreifenmarkierungen reagiert wird, sondern auch auf wahrgenommene stationäre Elemente der Szene, wie die Elemente einer Arbeitsstelle. Diese Funktionalität ist nach Kacher (2016) ebenfalls in der aktuellen E-Klasse verbaut. Zum Startpunkt des Forschungsprojekts war das System in der Serienfunktion des Versuchsträgers noch nicht enthalten und stellte damit einer Erweiterung zum Fahrstreifenhalteassistenten dar.

Das System wird aktiv, wenn der zur Verfügung stehende Fahrstreifen seitlich eingeengt ist. Einengungen der Straße entstehen beispielsweise durch Arbeitsstellen oder durch parkende Fahrzeuge am Straßenrand. Besonders in Arbeitsstellen kommt es nach Bakaba u. a. (2012, S.13,19) oft zu „Spiegelunfällen“ im Längsverkehr, wo es zu Zusammenstößen mit anderen Verkehrsteilnehmern kommt, welche seitlich versetzt in gleicher Richtung fahren. Dies bestätigt den Bedarf einer erweiterten Querführungsassistenz in Engstellen.

Der Engstellenassistent wurde im Rahmen einer bilateralen Beauftragung mit der Konzernforschung der *Volkswagen AG* für das Forschungsprojekt *UR:BAN (Urbaner Raum: Benutzergerechte Assistenzsysteme und Netzmanagement)*<sup>2</sup> entwickelt. Im Rahmen der Unterbeauftragung wurde vom Autor der vorliegenden Arbeit die Simulation als Test- und

---

<sup>1</sup>Die Idee der systematischen Auswertung wurden vom Autor der vorliegenden Arbeit erbracht. Die Koautoren haben durch intensive Diskussionen und Reviews zur Reife und Detaillierung des Ansatzes beigetragen. Die Realisierung des Ansatzes in dieser Form wäre ohne die Diskussionen nicht möglich gewesen.

<sup>2</sup>Homepage des UR:BAN-Projekts: <http://www.urban-online.org>, abgerufen 19.06.2016

Entwicklungswerkzeug im Projekt integriert und der modulare virtuelle Testbaukasten als systematisches Testkonzept entwickelt.

### 12.1.2 Funktionale Systemarchitektur

Die funktionale Systemarchitektur des Engstellenassistenten kann durch die funktionale Systemarchitektur aus Kapitel 3.2 dargestellt werden. Abbildung 12.1 zeigt durch die Einfärbung, welche Module der Architektur für den Engstellenassistenten umgesetzt wurden. Die einzelnen Module und deren Funktionen werden im Folgenden erläutert.

**Sensorik:** Die Umfeldsensorik besteht aus einer Stereokamera mit einem Öffnungswinkel von 42°. Das Steuergerät der Kamera führt eine Merkmalsextraktion auf Basis der aufgenommenen Bilder durch. Dabei werden die Merkmale im Bild in Form von Segmenten extrahiert, wobei aus jedem Bilderpaar 249 Segmente erstellt werden. Jedes Segment beinhaltet die Informationen über die relative Position und Geschwindigkeit in x-, y- und z-Richtung zum Ego-Fahrzeug sowie die Standardabweichungen der Werte. Weiterhin wird die Größe, Breite und Existenzwahrscheinlichkeit der Segmente ausgegeben. Eine Klassifikation auf stationäre oder dynamische Segmente wird von der Kamera nicht vorgenommen. Als weitere Umfeldsensoren wurden auf jeder Seite des Fahrzeugs jeweils zwei Ultraschallsensoren verbaut, um den Abstand zu parallel fahrenden Fahrzeugen zu messen.

Von der Fahrzeugsensorik werden die Raddrehzahlsensoren verwendet, um eine bordautonome Odometrie für eine lokale Eigenbewegungsschätzung zu realisieren. Des Weiteren werden das aktuelle Lenkmoment des Fahrers und der Lenkwinkel erfasst.

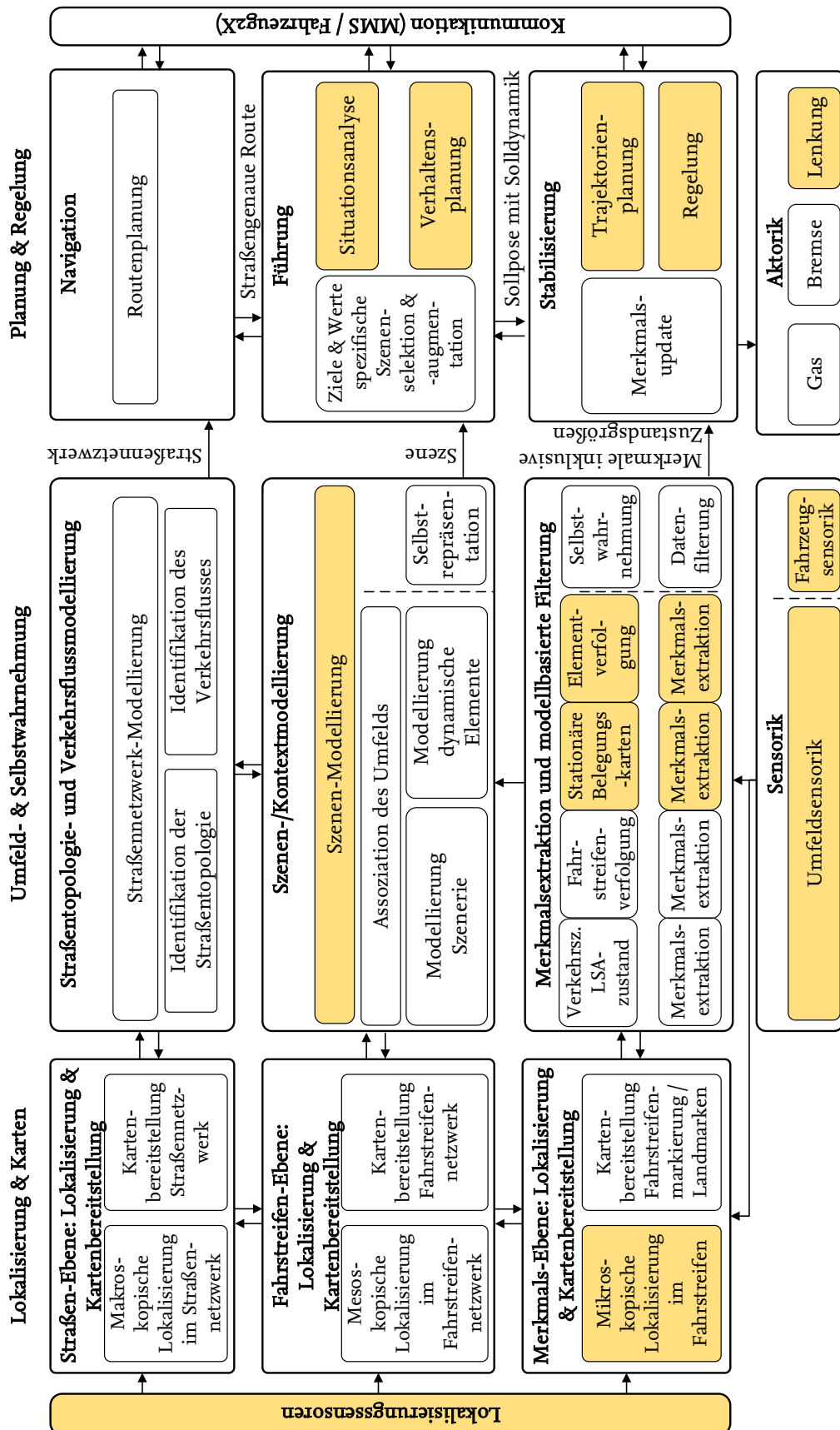
**Lokalisierungssensoren:** Für eine hochgenaue mikroskopische Lokalisierung, mit einer Abweichung von wenigen Zentimetern, wird eine Inertialplattform gekoppelt mit einem DGPS<sup>3</sup> verwendet.

**Lokalisierung und Karten:** Eine Karte wird von der Fahrfunktion des Engstellenassistenten nicht benötigt, da die Funktion lediglich auf die Elemente reagiert, die durch die Umfeldsensorik wahrgenommen werden. Eine eigene mikroskopische Lokalisierung wurde im Projekt nicht implementiert. Daher wurde die berechnete Positionslösung der Inertialplattform übernommen, um eine globale Position des Ego-Fahrzeugs zu bestimmen. Für die Fahrfunktion wird die globale Position nicht benötigt, da für diese die lokale Lokalisierung mittels Odometrie ausreicht. Die globale hochgenaue Position wird benötigt, um die Simulation als Test- und Entwicklungswerkzeug zu verwenden.

**Merkmalsextraktion und modellbasierte Filterung:** In dem Modul der Merkmalsextraktion und modellbasierten Filterung werden die Stereosegmente als stationär oder dynamisch klassifiziert. Hierzu werden die Segmente geclustert und über die Zeit verfolgt. Scholz u. a. (2014) beschreiben den Vorgang der Klassifikation der Segmente. Stationäre Segmente werden durch die in Scholz u. a. (2014) beschriebenen Heuristiken in eine stationäre Belegungskarte eingetragen. Die als dynamisch klassifizierten Segmente werden als Objekthypothesen über die Zeit verfolgt und in eine Objektliste übertragen. Dem Modul der Szenen-/Kontextmodellierung wird die stationäre Belegungskarte und die Objektliste übergeben.

**Szenen-/Kontextmodellierung:** Im Modul der Szenen-/Kontextmodellierung wird eine auf den Engstellenassistenten angepasste Szenenmodellierung durchgeführt. Dazu werden

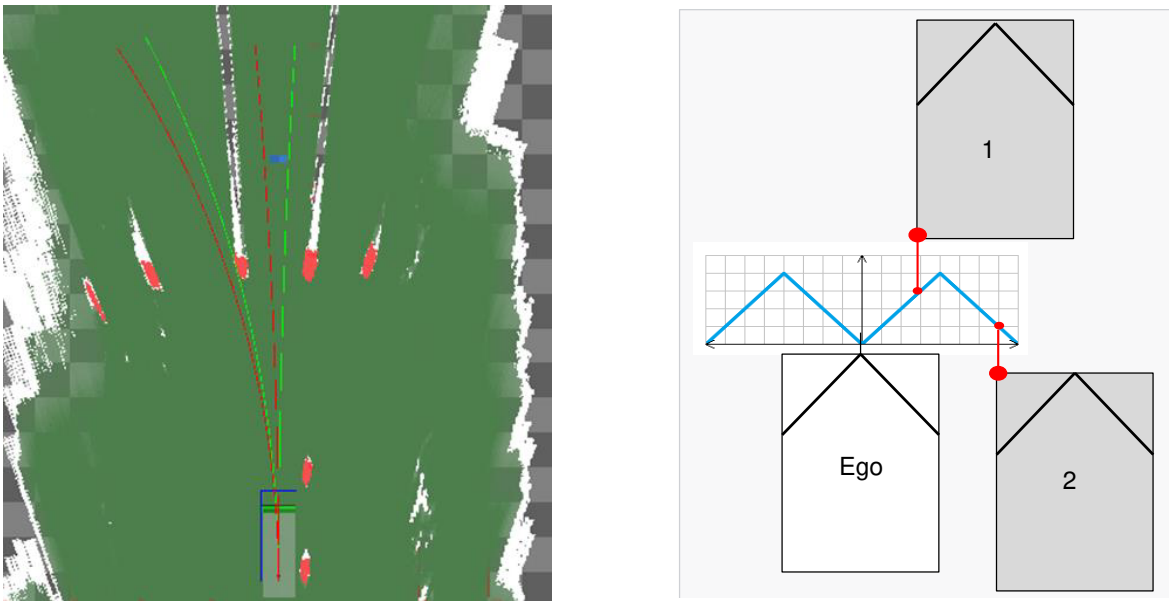
<sup>3</sup>DGPS: Differentielles globales Positionierungssystem



**Abbildung 12.1:** Funktionale Systemarchitektur des Engstellenassistenten nach der angepassten Architektur aus Kapitel 3.2. Die gelb markierten Module wurden im Rahmen des Projekts umgesetzt.



befahrbare Korridore aus der stationären Belegungskarte extrahiert. Ein Korridor wird auf der linken und rechten Seite durch einen Krümmungsrand begrenzt. Abbildung 12.2 zeigt das Ergebnis der Extraktion mit zwei möglichen Korridoren, jeweils begrenzt durch einen linken (rot) und rechten (grün) Krümmungsrand. Die befahrbaren Korridore werden der Szene hinzugefügt. Damit wird dem Modul der Führung eine Szene übergeben, die die befahrbaren Korridore sowie die Liste der dynamischen Objekte beinhaltet.



**Abbildung 12.2:** Darstellung der extrahierten Korridore (links) und Auswahl des relevanten Objekts und deren Referenzpunkt (rechts) (Scholz u. a., 2014)

**Führung:** Das Führungsmodul realisiert eine Situationsanalyse und Verhaltensplanung. Die Situationsanalyse vergleicht die befahrbaren Korridore mit dem Querführungswunsch des Fahrers, welcher aus dem Lenkradwinkel abgeleitet wird. Sollte der Querführungswunsch des Fahrers innerhalb eines befahrbaren Korridors liegen, gibt die Verhaltensplanung die durch den Fahrerwunsch vorgegebene Sollpose aus. Die Regelung wird in diesem Fall nicht aktiviert. Sollte der Querführungswunsch des Fahrers außerhalb der befahrbaren Korridore liegen, wird die Sollpose auf den Bereich des entsprechenden befahrbaren Korridors begrenzt. Weiterhin wird für die Situationsanalyse und Verhaltensplanung die Objektliste ausgewertet. Dazu werden nach der in Abbildung 12.2 dargestellten Heuristik das relevante Objekt und deren Referenzpunkt bestimmt. Der Referenzpunkt des ausgewählten Objekts schränkt die befahrbaren Korridore gegebenenfalls weiter ein, wenn sich der Referenzpunkt des Objekts im befahrbaren Korridor befindet.

**Stabilisierung:** Auf der Stabilisierungsebene wird eine Trajektorienplanung realisiert, die auf eine Bahnplanung reduziert wird, da keine Regelung der Längsdynamik stattfindet. Aus der Sollpose wird der Solllenkwinkel berechnet, der für die Erreichung der Sollpose benötigt wird. Wenn eine Differenz zwischen Soll- und Istlenkwinkel vorhanden ist, wird durch die Regelung unter Berücksichtigung des Fahrerlenkmoments ein Lenkmoment als Stellgröße zur Korrektur der Regelabweichung an die Lenkung übergeben.

**Aktorik:** Als Aktorikkomponente wird für die Realisierung der Querführungsassistentz die Lenkung verwendet. Die Lenkung übernimmt das Lenkmoment aus dem Regelungsmodul, wodurch der Solllenkwinkel geregelt wird. Weitere Aktorikkomponenten werden in dieser Ausbaustufe des Engstellenassistenten nicht verwendet.

## 12.2 Auswahl und Analyse von Einflussparametern

Das folgende Kapitel beschreibt die Anwendung der Auswahl und Analyse von Einflussparametern für das Fallbeispiel des Engstellenassistenten.

### 12.2.1 Auswahl von Einflussparametern

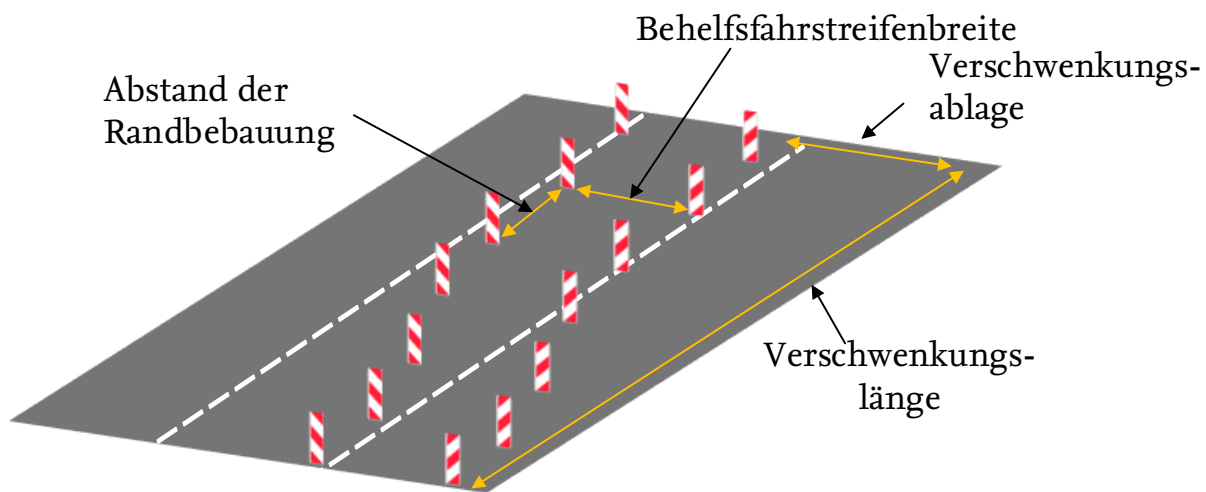
Für die Auswahl von Einflussparametern wurden zunächst verfügbare Informationsquellen gesichtet. Dabei wurde festgestellt, dass für den Engstellenassistenten zum Zeitpunkt der Implementierung die Funktionsspezifikation nur bedingt vorhanden war. Die Funktionsspezifikation bestand aus Anwendungsszenarien, die spezifiziert wurden und nach denen die Assistentenfunktion entwickelt wurde. Die Anwendungsszenarien wurden aus verfügbaren Normen und Richtlinien für Arbeitsstellen in Deutschland entnommen. Zur Bestimmung der geometrischen Maße des Versuchsträgers wurden Fahrzeugkataloge verwendet. Weiterhin wurden Experten befragt, die für die Planung und den Aufbau von Arbeitsstellen verantwortlich sind. Zusätzlich wurden Messfahrten durch reale Arbeitsstellen vorgenommen. Die Videodaten der Messfahrten wurden verwendet, um einen qualitativen Vergleich zwischen existierenden Arbeitsstellen und verfügbaren Verkehrszeichenplänen durchzuführen. Dabei wurden als exemplarisches Beispiel Arbeitsstellen auf der Autobahn A1 zwischen Bremen und Hamburg besichtigt, da für diese Arbeitsstellen die entsprechenden Verkehrszeichenpläne vorlagen. Durch den qualitativen Vergleich der Position von Verkehrszeichen und Verschwenkungen wurde abgeschätzt, dass sich der reale Aufbau der Arbeitsstelle am geplanten Aufbau orientiert. Damit wurde weiterhin abgeschätzt, dass die Normen und Richtlinien bei der Planung von Arbeitsstellen beachtet werden und somit zur Auswahl und Analyse der Einflussparameter verwendbar sind.

Damit standen als Informationsquellen die Normen und Richtlinien, Messfahrten, Fahrzeugkataloge und Expertenwissen für die Auswahl von Einflussparametern zur Verfügung. Die übrigen Quellen wurden zur Auswahl von Einflussparametern nicht verwendet. Tabelle 12.1 stellt dies dar.

Informationsquelle	Verfügbar
Funktionsspezifikation und Anwendungsszenarien	✗
4-Ebenen-Modell	✗
Normen und Richtlinien	✓
Fahrmanövertheorien	✗
Unfallanalysen	✗
Expertenwissen	✓
Messfahrten	✓
Fahrzeugkataloge	✓

**Tabelle 12.1:** Übersicht der verfügbaren Informationsquellen für die Auswahl von Einflussparametern. Legende: ✓: verfügbare Informationsquelle, ✗: nicht verfügbare Informationsquelle.

In den Normen und Richtlinien für die Sicherung von Arbeitsstellen (BMVI, 2009) werden verschiedene Typen von Arbeitsstellen als Regelpläne definiert. Je nach Arbeitsstellentyp und Straßendomäne beschreiben die Richtlinien Werte für die Parameter zum Aufbau der Arbeitsstellen. Neben den Regelplänen wurden Verkehrszeichenpläne gesichtet. Diese stellen die Anpassungen der Regelpläne durch Planungsexperten an die realen Gegebenheiten vor Ort dar<sup>4</sup>. Für den Test des Systems wurde das in Abbildung 12.3 dargestellte Szenario einer einstreifigen Verschwenkung ausgewählt. Die Verschwenkung beschreibt ein Standardszenario in Arbeitsstellen, da der Verkehr durch die Verschwenkung an der Arbeitsstelle vorbeigeführt wird.



**Abbildung 12.3:** Verschwenkung des Fahrstreifens in einer Arbeitsstelle durch Leitbaken

Eine Verschwenkung wird durch die Parameter der Verschwenkungslänge und der Verschwenkungsablage beschrieben. Aus den beiden Größen wird das Verschwenkungsmaß bestimmt, welches in den Normen und Richtlinien angegeben ist. Das Verschwenkungsmaß wird mit der Formel 12.1 berechnet.

$$\text{Verschwenkungsmaß} = \frac{\text{Verschwenkungsablage}}{\text{Verschwenkungslänge}} \quad (12.1)$$

Angegeben wird das Verschwenkungsmaß beispielsweise als 1:10. Dies bedeutet, dass bei einer Verschwenkungsablage von 10 Meter die Verschwenkung mindestens eine Länge von 100 Meter haben muss. Daneben definieren die folgenden Parameter eine Verschwenkung: die Behelfsfahrstreifenbreite, welche durch den lateralen Abstand der Randbebauung gemessen wird, die Art der Randbebauung und deren longitudinaler Abstand. Die empfohlenen Werte der Parameter sind in den Normen und Richtlinien zur Absicherung von Arbeitsstellen (BMVI, 2009) zu finden.

### 12.2.2 Analyse von Einflussparametern

Nach der Auswahl der Parameter wird im nächsten Schritt die Analyse der Parameter durchgeführt. Hierbei werden Diskretisierungsstufen für die verschiedenen Einflussparameter

<sup>4</sup>[https://mobil.hessen.de/irj/HSVV\\_Internet?cid=a6f243aaff7aedf0b2e4ff094c63e689](https://mobil.hessen.de/irj/HSVV_Internet?cid=a6f243aaff7aedf0b2e4ff094c63e689), abgerufen: 09.03.2016

abgeleitet und in das Schema eingetragen, welches in Kapitel 7.5 vorgestellt wurde. Als Informationsquelle steht das Wissen von Experten zur Verfügung. Bei den Experten handelt es sich zum einen um Mitarbeiter/innen von Firmen, die Arbeitsstellen aufbauen (Brückner, 2011), und zum anderen um Mitarbeiter/innen der niedersächsischen Landesbehörde für Straßenbau und Verkehr (Zulauf, 2011), die den Aufbau von Arbeitsstellen planen. Ein Optimierungsalgorithmus zur Identifikation von Diskretisierungsstufen oder eine statistische Auswertung von Straßen beziehungsweise Arbeitsstellen aus Online-Quellen stehen nicht zur Verfügung. Damit ergeben sich die in Tabelle 12.2 dargestellten verfügbaren Quellen zur Analyse der Einflussparameter.

Informationsquelle	Verfügbar
Optimierungsalgorithmen	✗
Statistische Auswertung	✗
Expertenwissen	✓

**Tabelle 12.2:** Übersicht der verfügbaren Informationsquellen für die Analyse von Einflussparametern. Legende: ✓: verfügbare Informationsquelle, ✗: nicht verfügbare Informationsquelle.

Im Folgenden werden für die identifizierten Einflussparameter die Diskretisierungsstufen herleitet.

**Typ der Randbebauung:** Für die Randbebauung der Verschwenkung werden bereits in den Normen und Richtlinien Diskretisierungen vorgenommen. Es werden drei Typen von Randbebauungen genannt. Dies sind Leitwände, Leitbaken und Leitkegel. Daher muss keine weitere Analyse und Diskretisierung vorgenommen werden.

**Abstand der Randbebauung:** Für den Abstand der Randbebauung werden je nach Domäne verschiedene longitudinale Abstände vorgeschlagen. Diese liegen zwischen 5 und 10 Meter. Für den Abstand der Randbebauung werden noch Schwankungen beim Aufbau beachtet. Daher wird der maximale Wert auf 11 Meter Abstand festgelegt. Dies entspricht einem Fehler von 10% beim Aufbau. Für die Diskretisierung wird eine lineare Funktion gewählt, bei der pro Diskretisierungsstufe der Abstand der Randbebauung um 2 Meter erhöht wird.

**Verschwenkungsmaß:** Für das Verschwenkungsmaß werden in den Normen und Richtlinien die Werte 1:10 und 1:20 genannt. Zwischen diesen Werten wird zusätzlich der Wert 1:15 als weitere Diskretisierungsstufe eingefügt, um eine detailliertere Aussage über den Einfluss des Verschwenkungsmaßes zu treffen.

**Verschwenkungsablage:** Für die Verschwenkungsablage werden in den Normen und Richtlinien keine Werte genannt. Für diesen Wert werden die Stufen so gewählt, dass sie Vielfache der Fahrstreifenbreite von 3,5 Meter entsprechen. Als maximale Verschwenkungsablage wird ein Wert von 14 Meter gewählt. Dies entspricht einer Verschwenkung von vier Fahrstreifen und bedeutet, dass bei einer dreistreifigen Autobahn der rechte Fahrstreifen der rechten Fahrbahn auf die Gegenfahrbahn verschwenkt wird.

**Behelfsfahrstreifenbreite:** Die Behelfsfahrstreifenbreite orientiert sich an der Mindestfahrstreifenbreite in Arbeitsstellen. Diese beträgt in der Regel bei einstreifigen Verschwenkungen mindestens 2,75 Meter (BMVI, 2009, S.89), damit auch Fahrzeuge, die eine Breite

größer als 2,00 Meter haben, durch die Arbeitsstelle fahren können. Jedoch kann unter Umständen auch bei einstreifigen Verschwenkungen eine Fahrstreifenbreite von 2,50 Meter auftreten. Dies ist beispielsweise der Fall, wenn während einer Arbeitsstelle eine Fahrbahn mit zwei Behelfsfahrstreifen getrennt wird und einer der Fahrstreifen auf die Gegenfahrbahn verschwenkt wird (BMVI, 2009, S.110). Als maximaler Wert wird 3,75 Meter angenommen. Dies entspricht dem maximalen Wert für die Fahrstreifenbreite in den geltenden Regelquerschnitten (FGSV, 2008, S.22f). Zwischen diesen Werten wird eine lineare Interpolation mit einer Diskretisierungsstufe von 0,25 Meter angenommen. Die Werte entsprechen damit den üblichen Fahrstreifenbreiten innerhalb von Arbeitsstellen (BMVI, 2009, S.89) und den Fahrstreifenbreiten der Regelquerschnitte auf Autobahnen (FGSV, 2008, S.22f).

Abbildung 12.4 zeigt das ausgefüllte Schema der Auswahl und Analyse der Einflussparameter für das Fallbeispiel der einstreifigen Verschwenkung einer Arbeitsstelle. Abhängigkeiten zwischen diesen Parametern und Diskretisierungsstufen wurden in dieser exemplarischen Fallstudie nicht modelliert.

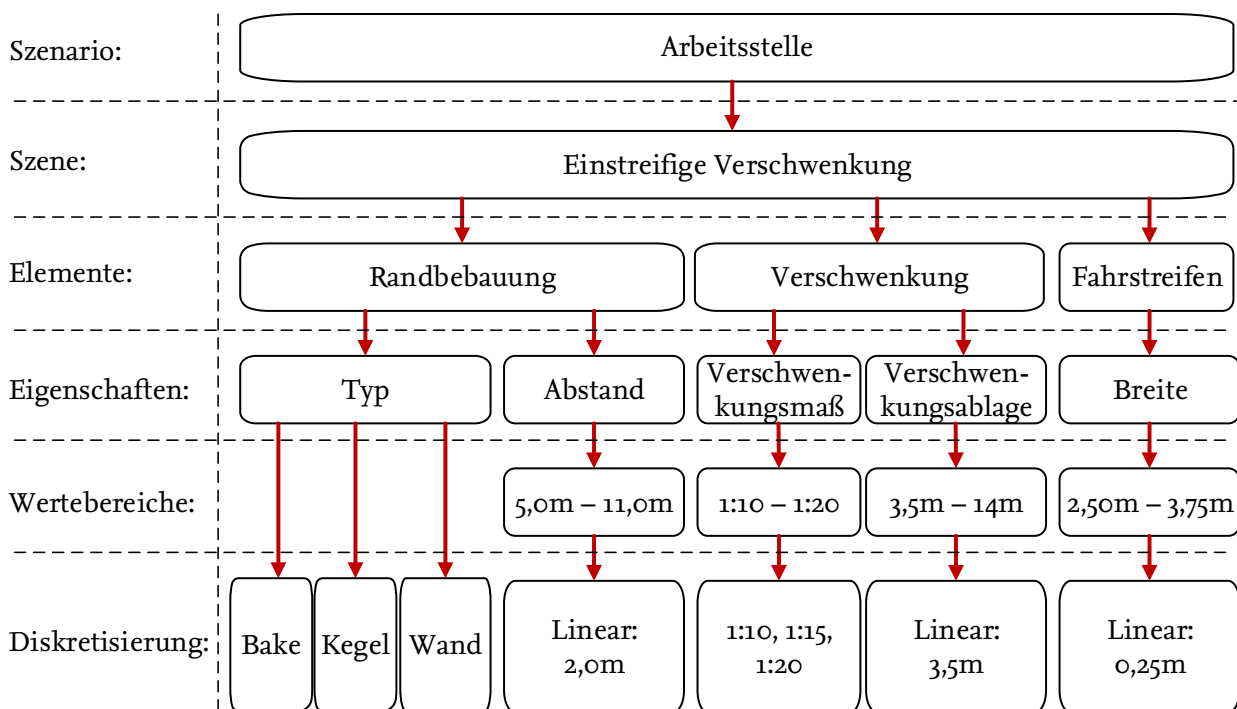


Abbildung 12.4: Anwendung des Schemas zur Auswahl und Analyse der Einflussparameter

## 12.3 Testfallerstellung

Das folgende Kapitel beschreibt die Anwendung der Testfallerstellung des modularen virtuellen Testbaukastens für das Fallbeispiel des Engstellenassistenten. Dafür wird die Anwendung des 4-Ebenen-Modells für die Erstellung von logischen Szenarien demonstriert. Aus den logischen Szenarien werden mittels kombinatorischer Algorithmen konkrete Testfälle abgeleitet. Die konkreten Testfälle werden der Testdurchführung übergeben.



### 12.3.1 Anwendung des 4-Ebenen-Modells

Zur Erstellung und Beschreibung der virtuellen Arbeitsstellen wird das in Kapitel 8.3 vorgestellte 4-Ebenen-Modell verwendet. Ziel der Anwendung ist eine hohe Variabilität der Szenarien. Dies bedeutet, dass sowohl das Basisstreckennetz als auch die Arbeitsstellen variabel beschreibbar sind. Damit müssen die identifizierten Einflussparameter inklusive der Wertebereiche im Modell abbildbar sein.

**1. Ebene - Basisstrecke:** Als Basisstrecken wurden reale Strecken auf dem Testgelände der Volkswagen AG in Ehra-Lessien sowie auf dem Testgelände der Audi AG in Neustadt vermessen. Dazu wurde ein teilautomatisierter Prozess entwickelt, um Streckenzüge ohne Kreuzungen für die Simulation im *OpenDrive*-Format aufzubereiten. Der erste Schritt besteht darin, die Strecke mit einem Fahrzeug abzufahren, welches mit einer hochgenauen Ortungsplattform ausgestattet ist. Für die Erstellung der Strecken im *OpenDrive*-Format wird eine Referenzlinie benötigt, an der sich der Verlauf der Fahrstreifen orientiert (Dupuis u. a., 2010b). Daher wurde für die Vermessung der Straße mit den linken Rädern auf der linken Markierung der Teststrecken gefahren, um diese Markierung der realen Straße als Referenzlinie für die virtuelle Strecke zu vermessen. Die hochgenauen Positionen des Fahrzeugs wurden verwendet, um mit einer kubischen Splineinterpolation die georeferenzierte Referenzlinie der virtuellen Strecke zu erstellen. Die Breite der Fahrstreifen wurde von Hand vermessen und mit 3,5 Meter als konstanter Wert im *OpenDrive*-Format abgelegt. Mit diesem Prozess besteht die Möglichkeit, reale Streckenabschnitte zu vermessen und diese als Basisstrecke für die Arbeitsstellen zu nutzen. Abbildung 12.5 zeigt eine Überlagerung der rekonstruierten Strecke für die Simulation mit der Teststrecke in Ehra-Lessien.

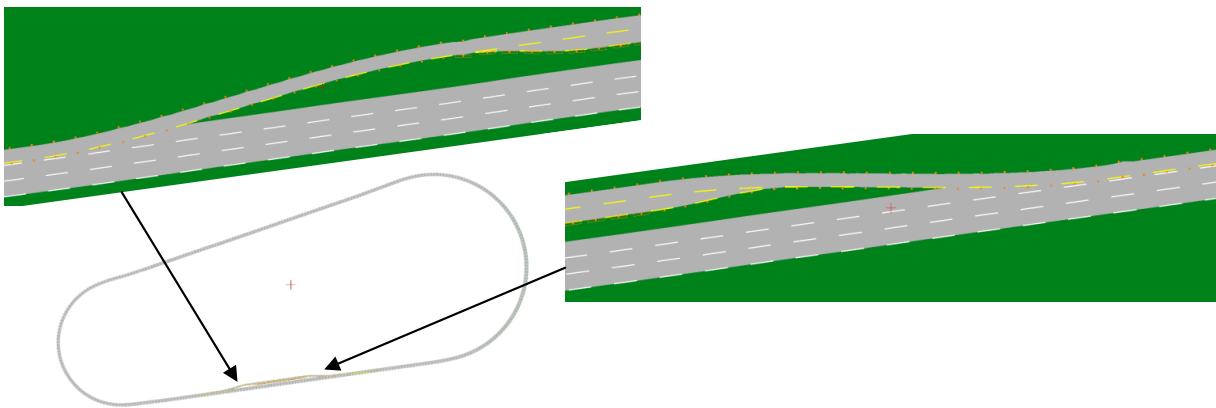


**Abbildung 12.5:** Überlagerung der rekonstruierten Strecke (blau) für die Simulation mit der realen Teststrecke. Bilder © 2015 Google, Kartendaten © GeoBasis-DE/BKG (© 2009), Google.

**2. Ebene - Situationsspezifische Anpassungen der Basisstrecke:** Der Aufbau der Arbeitsstellen wird als situationsspezifische Anpassung der Basisstrecke auf der zweiten



Ebene des 4-Ebenen-Modells realisiert. Ein proprietäres XML-Format<sup>5</sup> beschreibt den Aufbau der Arbeitsstelle. Dieses definiert die Arbeitsstelle relativ zur Basisstrecke, wodurch die Arbeitsstelle an beliebigen Stellen der Basisstrecke positionierbar ist. Dies zeigt den Vorteil des hierarchischen 4-Ebenen-Modells, da durch die Änderung der Basisstrecke die gleiche Arbeitsstelle auch auf anderen Basisstrecken aufgebaut und getestet werden kann. Im Beschreibungsformat sind beliebig viele aufeinanderfolgende Verschwenkungen inklusive beschreibender Parameter definierbar. Damit wurde die Möglichkeit geschaffen, ein- oder mehrstreifige Verschwenkungen zu beschreiben. Im 4-Ebenen-Modell wird mittels der Basisstrecke, welche im *OpenDrive*-Format vorliegt, und der Arbeitsstellenbeschreibung eine angepasste Beschreibung der Szenerie im *OpenDrive*-Format erstellt. Diese enthält die Basisstrecke inklusive der situationsspezifischen Anpassungen, die in diesem Fall durch die Arbeitsstellen realisiert sind. Abbildung 12.6 zeigt einen Ausschnitt der Basisstrecke inklusive erstellter Arbeitsstelle. Es wurde eine einstreifige Verschwenkung nach links, gefolgt von einem zweistreifigen Abschnitt und einer einstreifigen Verschwenkung nach rechts definiert.



**Abbildung 12.6:** Erweiterte Basisstrecke um die Arbeitsstelle

**3. Ebene - Akteure:** Auf der dritten Ebene wird das Verhalten der Akteure im Szenario definiert. Für die Szenarien des Engstellenassistenten besteht die Möglichkeit, das dynamische Verhalten von Lastkraftwagen innerhalb der Arbeitsstelle zu beschreiben. Das Verhalten der Lastkraftwagen wird durch die folgenden Parameter beschrieben: Lastkraftwagen mit oder ohne Anhänger, Startpunkt relativ zur Arbeitsstelleneinfahrt sowie die relative Geschwindigkeit zum Ego-Fahrzeug. Zusätzlich können bei mehrstreifigen Arbeitsstellen Fahrstreifenwechsel der Lastkraftwagen getriggert werden, sodass während der Tests Überholmanöver durchgeführt werden können.

**4. Ebene - Umweltbedingungen:** Auf der vierten Ebene des Modells werden die Umweltbedingungen definiert. Diese Ebene wird für den Test des Engstellenassistenten nicht definiert, da keine wetter- oder lichtabhängigen Sensormodelle zur Verfügung stehen.

### 12.3.2 Anwendung der systematischen Testfallableitung

Auf Basis der Auswahl und Analyse der Einflussparameter und der logischen Szenarien werden im Folgenden mittels der systematischen Testfallableitung konkrete Testfälle erstellt.

<sup>5</sup>XML ist die Abkürzung für Extensible Markup Language, die frei übersetzt „erweiterbare Auszeichnungssprache“ bedeutet.

Dafür werden Äquivalenzklassen gebildet und eine Grenzwertbetrachtung durchgeführt. Anschließend werden mit der kombinatorischen Testfallableitung Testfälle erstellt.

**Äquivalenzklassenbildung und Grenzwertbetrachtung:** Für die Äquivalenzklassenbildung und Grenzwertbetrachtung werden die identifizierten Diskretisierungsstufen verwendet. Für jede Diskretisierungsstufe wird eine Äquivalenzklasse gebildet. Da derzeit keine Aussage über das Verhalten des Systems auf den jeweiligen Diskretisierungsstufen getroffen werden kann, besteht keine Möglichkeit, die Diskretisierungsstufen in Äquivalenzklassen zusammenzulegen. Mittels der Grenzwertbetrachtung werden zusätzliche Äquivalenzklassen für das Verschwenkungsmaß und den Abstand der Randbebauung hinzugefügt. Für das Verschwenkungsmaß werden die Stufen 1:25 sowie 1:7 und 1:5 hinzugefügt. Durch die Einführung der Stufen 1:5 und 1:7 sollen die Grenzen der Fahrfunktion gezeigt werden. Dabei werden keine Nebenbedingungen zwischen den Parametern oder zu weiteren Informationsquellen beachtet, sodass womöglich für das Testobjekt nicht zu bestehende Szenarien generiert werden. Allerdings besteht so die Möglichkeit, zu untersuchen, unter welchen Bedingungen Testfälle bestanden oder nicht mehr bestanden werden. Für den Abstand der Randbebauung wird die Stufe von 13 Meter Abstand hinzugefügt. Dadurch wird getestet, wie das Assistenzsystem bei Abständen außerhalb der Normen und Richtlinien funktioniert. Die folgende Tabelle 12.3 zeigt die für den Test ausgewählten Diskretisierungsstufen der Einflussparameter.

Einflussparameter	Diskretisierungsstufen					
Verschwenkungsmaß	1:5	1:7	1:10	1:15	1:20	1:25
Verschwenkungsablage in m	3,5	7,0	10,5	14,0		
Behelfsfahstreifenbreite in m	2,5	2,75	3,0	3,25	3,5	3,75
Abstand der Randbebauung in m	5,0	7,0	9,0	11,0	13,0	
Typ der Randbebauung	Leitbake	Leitkegel	Leitwand			

**Tabelle 12.3:** Übersicht der ausgewählten Einflussparameter und Diskretisierungsstufen des stationären Arbeitsstellenszenarios

**Kombinatorische Testfallableitung:** Mit den ausgewählten Diskretisierungsstufen werden durch den kombinatorischen Algorithmus *IPOG* (Lei u. a., 2008) konkrete Testfälle abgeleitet. Als Testabdeckung wird eine paarweise Abdeckung gewählt. Damit lassen sich die Fehler identifizieren, die von zwei Parameterstufen abhängig sind. Nach Kuhn u. a. (2004) sind dies im Schnitt 70% der Fehler im System. Fehler, die von drei oder mehr Parametern abhängen, werden mit diesem Versuchsplan nur zum Teil gefunden, da nur ein Teil der höheren Testabdeckung von den abgeleiteten Testfällen abgedeckt wird. Bei einem Test aller möglichen Kombinationen müssten 2160 Testfälle durchgeführt werden. Durch die paarweise Abdeckung wird die Anzahl der benötigten Testfälle auf 36 reduziert. Mit der Durchführung dieser reduzierten Anzahl von Testfällen wird analysiert, welche Effekte der Einflussparameter auf die Bewertungskriterien identifizierbar sind. Allerdings ist zu beachten, dass während der Auswahl und Analyse keine Abhängigkeiten zwischen den Parametern und Diskretisierungsstufen definiert wurden. Dadurch werden diese auch nicht bei der

kombinatorischen Testfallableitung betrachtet, wodurch die Herausforderung besteht, dass unter gegebenen technischen Randbedingungen, wie ein maximales Lenkmoment oder einer maximal zulässigen Querbeschleunigung, nicht zu bestehende Testfälle abgeleitet werden. Dies gilt insbesondere für die Grenzwerte, die im vorherigen Schritt hinzugefügt wurden.

Die folgende Tabelle 12.4 zeigt die Testfälle, welche mittels der kombinatorischen Testfallableitung mit einer paarweisen Testabdeckung erstellt wurden. Hier zeigt sich die oben beschriebene Herausforderung der erstellten Testfälle. Beispielsweise wird beim Testfall 1 durch das Verschwenkungsmaß von 1:5 und einer Verschwenkungsablage von 3,5 Meter von der Fahrfunktion gefordert, dass diese durch Verschwenkungen mit hohen Krümmungen fahren kann. Daher können derzeit gegebenenfalls durch die ungünstige Kombination von Diskretisierungsstufen nicht zu bestehende Testfälle abgeleitet werden. Diese wurden in der Testdurchführung dennoch durchgeführt, um die Grenzen der Fahrfunktion zu analysieren und zu zeigen, unter welchen Bedingungen die Fahrfunktion Testfälle besteht oder nicht mehr besteht.

Testfall	Verschwenkungsmaß	Verschwenkungsablage in Meter	Behelfsfahrstreifenbreite in Meter	Abstand der Randbebauung in Meter	Art der Randbebauung
1	1:5	3,5	2,50	5	Bake
2	1:5	7,0	2,75	7	Kegel
3	1:5	10,5	3,00	9	Wand
4	1:5	14,0	3,25	11	Bake
5	1:5	3,5	3,50	13	Kegel
6	1:5	7,0	3,75	5	Wand
7	1:7	10,5	2,50	7	Bake
8	1:7	14,0	2,75	9	Kegel
9	1:7	3,5	3,00	11	Wand
10	1:7	7,0	3,25	13	Bake
11	1:7	10,5	3,50	5	Kegel
12	1:7	14,0	3,75	7	Wand
13	1:10	7,0	2,50	9	Bake
14	1:10	10,5	2,75	11	Kegel
15	1:10	14,0	3,00	13	Wand
16	1:10	3,5	3,25	5	Kegel
17	1:10	3,5	3,50	7	Bake
18	1:10	3,5	3,75	9	Bake
19	1:15	7,0	2,50	11	Wand
20	1:15	10,5	2,75	13	Bake
21	1:15	14,0	3,00	5	Kegel
22	1:15	10,5	3,25	7	Wand
23	1:15	7,0	3,50	9	Wand
24	1:15	3,5	3,75	11	Kegel
25	1:20	14,0	2,50	13	Kegel
26	1:20	3,5	2,75	5	Wand

Testfall	Verschwenkungsmaß	Verschwenkungsablage in Meter	Behelfsfahrstreifenbreite in Meter	Abstand der Randbebauung in Meter	Art der Randbebauung
27	1:20	7,0	3,00	7	Bake
28	1:20	10,5	3,25	9	Wand
29	1:20	14,0	3,50	11	Bake
30	1:20	10,5	3,75	13	Kegel
31	1:25	7,0	2,50	5	Bake
32	1:25	14,0	2,75	7	Kegel
33	1:25	3,5	3,00	9	Wand
34	1:25	10,5	3,25	11	Kegel
35	1:25	10,5	3,50	13	Kegel
36	1:25	14,0	3,75	13	Bake

**Tabelle 12.4:** Reduzierter Testplan mit einer paarweisen Testabdeckung

## 12.4 Testdurchführung

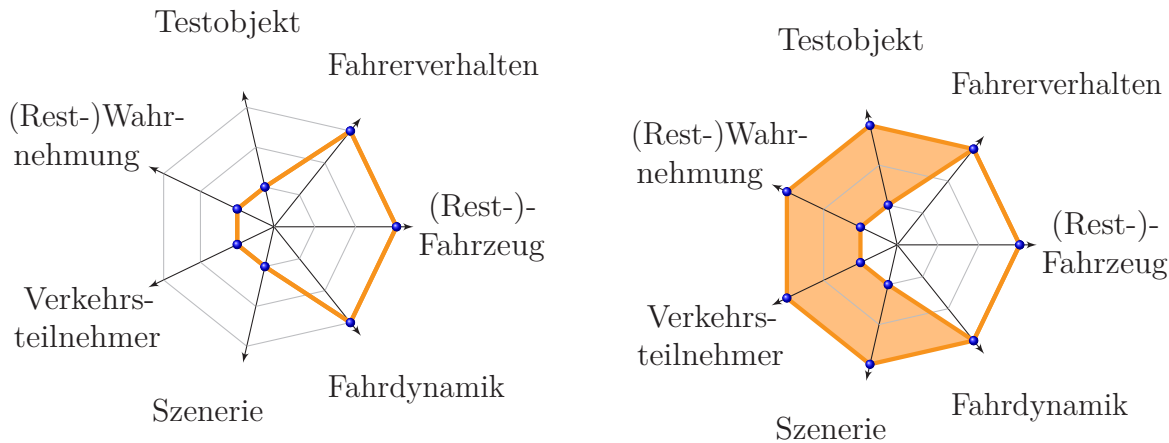
Dieses Unterkapitel beschreibt die Anwendung der Testdurchführung des modularen virtuellen Testbaukastens im Projekt des Engstellenassistenten. Dazu wird zunächst die Zuordnungsmethode für Testfälle angewendet. Anschließend werden die entwicklungsbegleitenden Testtreiber und die Durchführung der kombinatorisch erstellten Testfälle vorgestellt.

### 12.4.1 Anwendung der Zuordnungsmethode

Mit den Testfällen sollen die Module der Merkmalsextraktion, der modellbasierten Filterung, der Szenen-/Kontextmodellierung, der Führung und der Stabilisierungsebene der Fahrfunktion des Engstellenassistenten getestet werden. Der Regler für die Querführung soll ebenfalls mit dem Verfahren ausgelegt werden, wodurch sich die Anforderung an eine reale Fahrdynamik ergibt. Damit besteht die Forderung, dass das X-in-the-Loop-Verfahren im Zielfahrzeug einsetzbar ist, damit eine hohe Ergebnisgüte bezüglich der Fahrdynamik erreicht wird. Da ein reales Zielfahrzeug eingesetzt werden soll, soll das Testverfahren ein reales Fahrerverhalten zur Verfügung stellen. Da der Fahrer das System stets überwachen muss, kann er so bei einem Fehlverhalten des Systems in die Querführung eingreifen.

Weiterhin besteht die Forderung, dass eine Vielzahl von Arbeitsstellen in einer kurzen Zeitspanne getestet werden sollen. Daher müssen die Szenerie und die übrigen Verkehrsteilnehmer als Simulation zur Verfügung gestellt werden, wodurch auch die Umfeldsensorik simuliert werden muss. Für die Szenen-/Kontextmodellierung werden Objektlisten der übrigen Verkehrsteilnehmer und eine wahrgenommene Szenerie in Form einer befüllten stationären Belegungskarte benötigt. Dadurch ergibt sich, dass die Simulation die Module der Merkmalsextraktion und modellbasierten Filterung darstellen muss. Für das X-in-the-Loop-Verfahren bedeutet dies, dass die (Rest-)Wahrnehmung durch eine Simulation zur Verfügung gestellt werden muss. Für die Erweiterung der Testschleife um die Merkmalsextraktion und modellbasierte Filterung werden simulierte Sensorrohdaten der verwendeten Umfeldsensorik benötigt. Diese müssen durch einen Testtreiber des X-in-the-Loop-Verfahrens zur Verfügung

gestellt werden. Es ergeben sich die in Abbildung 12.7 dargestellten Anforderungen an die X-in-the-Loop-Verfahren.



Anforderungen des Testfalls, dargestellt durch ein Kiviat-Diagramm

Kiviat-Diagramm des *Vehicle-in-the-Loop*

**Abbildung 12.7:** Vergleich der Anforderung des Testfalls mit dem X-in-the-Loop-Verfahren *Vehicle-in-the-Loop*

Durch einen direkten Vergleich der Kiviat-Diagramme der klassifizierten X-in-the-Loop-Verfahren aus Kapitel 4.3.3 lässt sich feststellen, dass derzeit für die betrachteten Tests nur das *Vehicle-in-the-Loop*-Verfahren verwendbar ist. Eine gewichtete Bewertung von verschiedenen Gütekriterien ist für eine weitere Auswahl nicht notwendig. Aufgrund dieser Zuordnung wurde für den Test des Engstellenassistenten ein *Vehicle-in-the-Loop* ähnlicher Versuchsaufbau erstellt, welcher im folgenden Kapitel erläutert wird.

### 12.4.2 Beschreibung des Versuchsträgers

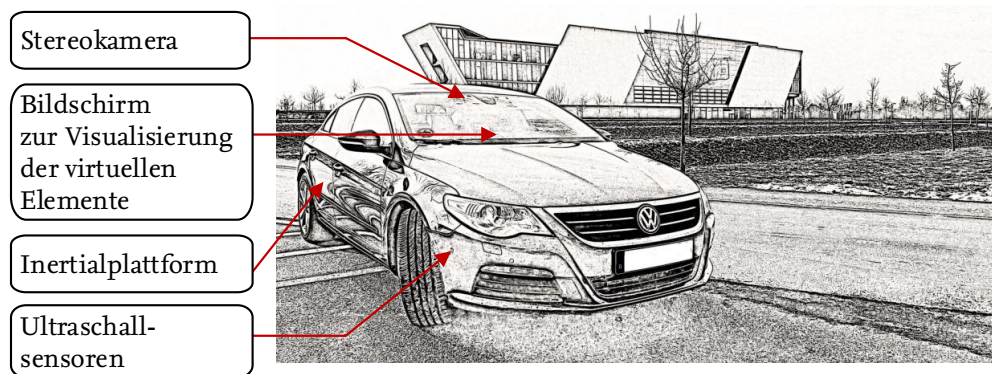
Als Versuchsträger wurde ein Passat CC eingesetzt. Dieser wurde mit weiterer Umfeldsensorik ausgestattet. In der Frontscheibe wurde eine Stereokamera verbaut. Seitlich wurden zu jeder Seite zwei Ultraschallsensoren integriert. Weiterhin wurde eine Inertialplattform inklusive DGPS-System verbaut, um eine hochgenaue Position zu erhalten.

Nach der Auswahl der Testwerkzeuge wurde für die Durchführung der Tests die Idee des *Vehicle-in-the-Loop* von Bock (2008) aufgegriffen und die Testumgebung einer erweiterten Realität (engl. augmented reality) aufgebaut. Der Unterschied zum *Vehicle-in-the-Loop* besteht darin, dass der Entwickler kein Helmdisplay (engl. head mounted display) während der Tests trägt, sondern auf einem Bildschirm in der Frontscheibe des Fahrzeugs ein überlagertes Bild der Kamera und der virtuellen Elemente sieht. Abbildung 12.8 zeigt eine Zeichnung des Versuchsträgers inklusive der Einbauorte der Umfeldsensorik und des Bildschirms zur Visualisierung der erweiterten Realität.

Der Testbaukasten erstellt mit Hilfe des 4-Ebenen-Modells und der systematischen Testfallableitung die konkreten Testfälle und überträgt diese in die Simulation. Die Streckendaten der ersten und zweiten Ebene des 4-Ebenen-Modells werden im *OpenDrive*-Format beschrieben.

<sup>6</sup>Mit der freundlichen Genehmigung zur Verwendung und Veränderung vom Fotografen Sven Chlosta.





**Abbildung 12.8:** Links: Zeichnung des Versuchsträgers inklusive der Einbauorte der Umfellsensorik und des Bildschirms zur Visualisierung der erweiterten Realität<sup>6</sup>.

Als Simulationsumgebung wird die Software *Virtual Test Drive* der Firma *Vires* eingesetzt. Die Beschreibung des Verhaltens der Akteure wurde in einem speziell für die Software benötigten Format gespeichert. Nach der Übertragung der Daten in die Simulation wird das Szenario gestartet.

Über die Inertialplattform wird kontinuierlich die Position des Fahrzeugs relativ zur vermessenen Strecke bestimmt. Diese Position wird der Simulationsumgebung übergeben, sodass die Position des Ego-Fahrzeugs und des virtuellen Ego-Fahrzeugs in der Simulation stets synchronisiert ist. Die virtuellen Verkehrsteilnehmer, die sich um das virtuelle Ego-Fahrzeug befinden, werden mittels Objektlisten den Testtreibern des modularen virtuellen Testbaukasten übergeben, welche im folgenden Kapitel beschrieben werden.

### 12.4.3 Entwicklungsbegleitende Testtreiber des Engstellenassistenten

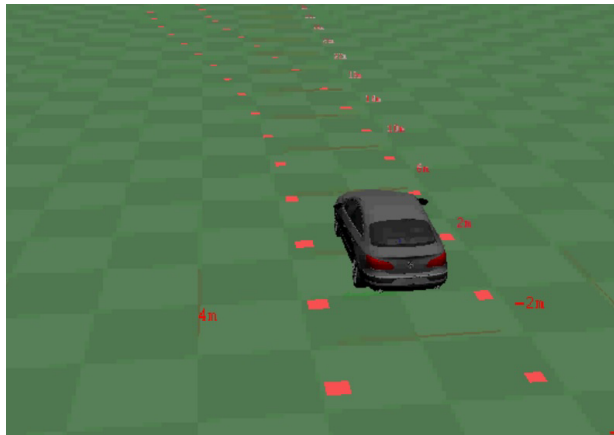
Für den Test des Engstellenassistenten wurden verschiedene Testtreiber implementiert. Diese werden im Folgenden erläutert.

#### Genaue Offline-Referenzbelegungskarte

Der erste Testtreiber besteht aus Algorithmen, die die Möglichkeit bieten, Verschwenkungen im *OpenDrive*-Format zu beschreiben. Die Arbeitsstelle wurden dabei in einem lokalen Koordinatensystem definiert. Darauf aufbauend wurden Algorithmen entwickelt, die die Elemente der Arbeitsstelle aus dem *OpenDrive*-Format auslesen und die stationären Elemente anhand der Position und deren Abmaßen in eine stationäre Belegungskarte eintragen. Damit wurden genaue Referenzbelegungskarten erstellt. Abbildung 12.9 zeigt eine solche genaue Referenzbelegungskarte einer Verschwenkung. Der Testtreiber stellt damit die Eingangsdaten für die Szenen- und Kontextmodellierung zur Verfügung.

Vorteil dieser Testmethode ist die nicht erforderliche Simulationsumgebung. Die Arbeitsstelle wird auf Basis des *OpenDrive*-Formats beschrieben und mittels der genauen Belegungskarte der Fahrfunktion zur Verfügung gestellt. Objektlisten oder Sensordaten aus der Simulation werden zur Laufzeit der Tests nicht benötigt. Weiterhin wird keine hochgenaue Ortung benötigt. Aufgrund der lokalen Beschreibung der Arbeitsstellen ist eine lokale Eigenbewegungsschätzung des Fahrzeugs für das Durchfahren der Arbeitsstelle ausreichend.





**Abbildung 12.9:** Genaue Referenzbelegungskarten einer erstellten Verschwenkung

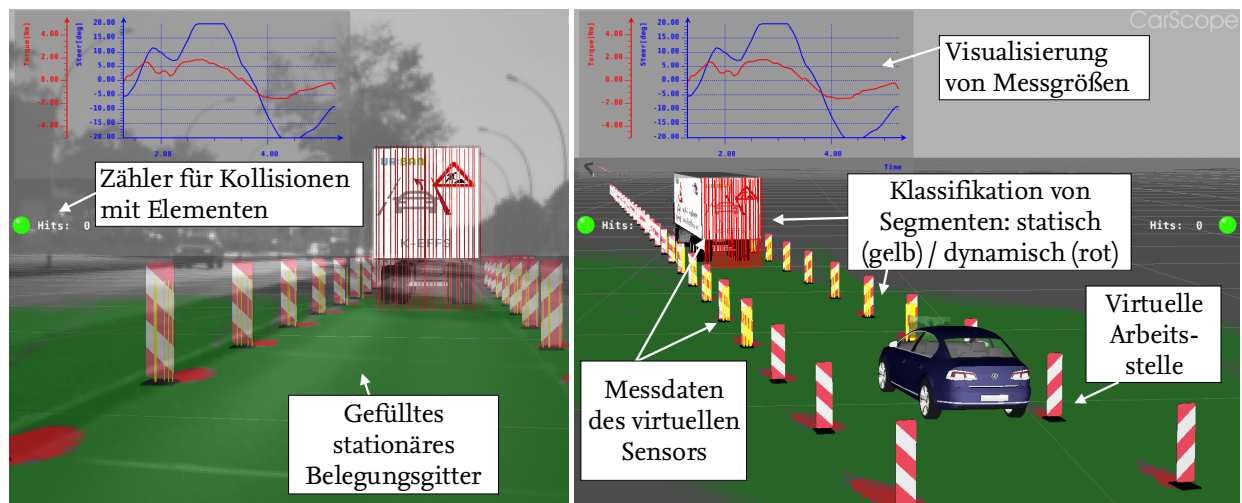
### Online erstellte Sensorinformationen

In einem erweiterten Testtreiber wurden die Merkmalsextraktion und die modellbasierte Filterung in die Testschleife integriert. Dafür wurden aus der Simulation die Sensordaten der Stereokamera und Objektlisten generiert, wodurch die Simulation zur Laufzeit der Tests im Fahrzeug benötigt wurde. Des Weiteren werden eine georeferenzierte Basisstrecke und eine hochgenaue Lokalisierung benötigt, um die Position des realen Fahrzeugs mit der Position des virtuellen Fahrzeugs zu synchronisieren. Die Simulation wird während der Tests mit der Bildrate der Kamera synchronisiert. Dies bedeutet, dass für jedes Bild der Kamera ein Simulationsschritt durchgeführt wird. Damit werden die Kamerabilder mit der Bewegung des Fahrzeugs und den Daten der Simulation synchronisiert.

Für die Erstellung der Sensordaten wurde ein konfigurierbares Strahlenmodell entwickelt, welches die Objektlisten der Simulationsumgebung als Eingangsdaten verwendet. Dabei handelt es sich um das Strahlenmodell, welches in Kapitel 9.4 vorgestellt wurde und von Lackmann (2015) weiterentwickelt wurde. Das Modell simuliert die Sichtstrahlen der Kamera. Für jedes Stereosegment, welches von der Kamera generiert wird, wurde ein Strahl im Strahlenmodell konfiguriert. Jedes Segment beinhaltet die Information, ob ein virtuelles Objekt getroffen wurde. Wenn ein Objekt getroffen wurde, werden die Informationen über den Treffpunkt, die Segmentbreite und -höhe sowie die Geschwindigkeit des Objekts in den Segmentinformationen gespeichert. Des Weiteren besteht die Möglichkeit die Informationen des Segments mit einem Rauschmodell zu überlagern, um neben den genauen Daten veräuschte Daten zur Verfügung zu stellen. Darüber hinaus besteht die Option das Segment als statisch oder dynamisch durch die Simulation zu klassifizieren. Dadurch wird die Information über die Klassifikation aus der Simulation generiert. In der Merkmalsextraktion kann damit die Dynamikklassifikation übersprungen werden, wodurch diese nicht getestet wird.

Die Liste mit virtuellen Segmenten wird in jedem Simulationsschritt mit der Segmentliste der Stereokamera verglichen. Dabei wird das Dichtere der beiden Segmente in eine fusionierte Segmentliste übertragen. Damit wird die Möglichkeit geschaffen, mit einer virtuellen beziehungsweise realen Arbeitsstelle sowie mit einer Arbeitsstelle mit gemischt virtuellen und realen Elementen zu testen. Die fusionierte Segmentliste wird im Anschluss der Merkmalsextraktion übergeben.

Abbildung 12.10 zeigt zwei Darstellungen der erweiterten Realität. Die linke Seite zeigt das überlagerte Bild aus der Perspektive der Kamera. Auf der rechten Seite ist der Aufbau als Übersichtsbild zu sehen. Neben dem Kamerabild und den virtuellen Elementen wird zusätzlich ein Graph zur Visualisierung von Messgrößen eingeblendet. Die Belegungskarte wurde durch die virtuellen stationären Segmente (gelbe Segmente) befüllt. Die als dynamisch klassifizierten Segmente (rote Segmente) werden nicht in die Belegungskarte eingetragen. Um eine erste Auswertung schon während der Tests zu ermöglichen, wurden Algorithmen zur Detektion von Kollisionen mit virtuellen Elementen implementiert. Die Anzahl der Kollisionen auf der linken und rechten Seite pro Test werden durch einen Zähler dargestellt.



**Abbildung 12.10:** Testumgebung mit erweiterter Realität des Engstellenassistenten.

Links: Sicht aus der Perspektive der Kamera, Rechts: Übersichtsbild

#### 12.4.4 Durchführung der kombinatorisch erstellten Testfälle

Für die Fallstudie werden die kombinatorisch abgeleiteten Testfälle durchgeführt. Dazu wird der Testtreiber für die online erstellten Sensorinformationen verwendet. Für die Erstellung der Sensordaten wird das beschriebene Strahlenmodell verwendet, um die virtuelle Arbeitsstellen in Form von Stereosegmenten wahrzunehmen. Dabei werden in jedem Simulationsschritt 249 Segmente erstellt.

Als Basisstrecke dient die erfasste und rekonstruierte Strecke auf dem Testgelände nahe Ehra-Lessien. Die virtuellen Arbeitsstellen werden auf dieser Basisstrecke errichtet und mit dem realen Fahrzeug durchfahren. Die virtuellen Elemente der Arbeitsstelle und die virtuellen Stereosegmente werden dem Fahrer auf einem Bildschirm visualisiert. Zur Vergleichbarkeit der Testfälle werden die Arbeitsstellen mit einer konstanten Geschwindigkeit von  $60 \frac{\text{km}}{\text{h}}$  durchfahren. Weiterhin wird für die Vergleichbarkeit der Testergebnisse das Assistenzsystem so konfiguriert, dass keine ständige Interaktion mit dem Fahrer notwendig ist. Damit ist es möglich, dass der Fahrer während der Durchfahrt die Hände nicht am Lenkrad hat und nur die Fahrfunktion einen Einfluss auf die Querregelung des Fahrzeugs besitzt.

Ziel der Fallstudie ist die Bewertung der kombinatorischen Testfallableitung. Da die Anzahl der Testfälle durch die Kombinatorik und der paarweisen Abdeckung von 2160 auf 36

Testfälle reduziert wurde, wird mittels der Testfallauswertung analysiert, welche Effekte der Einflussparameter auf die Bewertungskriterien messbar sind.

## 12.5 Testfallauswertung

Im Folgenden wird die Anwendung der Testfallauswertung in der Fallstudie des Engstellenassistenten beschrieben. Dazu werden zunächst Bewertungskriterien und Qualitätsverlustfunktionen für das System definiert. Anschließend findet eine Analyse der Testergebnisse durch die Methode der Mittelwertanalyse statt.

### 12.5.1 Bewertungskriterien

Die Bewertungskriterien wurden auf Basis von Sicherheits- und Komfortaspekten ausgewählt. Die folgenden Bewertungskriterien wurden definiert.

**Minimaler Abstand zur Randbebauung:** Dieses Bewertungskriterium beschreibt den minimalen Abstand zur Randbebauung während der Fahrt durch die Verschwenkung. Der Abstand wird mittels der euklidischen Distanz zwischen den Konturen des Versuchsfahrzeugs und den virtuellen Modellen der Randbebauung bestimmt. Als Qualitätsverlustfunktion wird die Maximierungsfunktion gewählt, da der Abstand zur Randbebauung möglichst groß sein soll. Für die Bestimmung des minimal erlaubten Werts wurde die geringste Fahrstreifenbreite in Arbeitsstellen, welche 2,50 Meter beträgt, mit der Breite des Versuchsträgers abgeschätzt. Der Versuchsträger besitzt eine Breite von 1,85 Meter (Volkswagen AG, 2011)<sup>7</sup>. Damit ergibt sich auf jeder Seite eine Distanz von ca. 0,32 Meter zwischen den Elementen der Arbeitsstelle und dem Versuchsträger, wenn der Versuchsträger in der Mitte des Behelfsfahrstreifens fährt. Der minimal erlaubte Wert wurde in empirischen Versuchen so abgeschätzt, dass ein Abstand von 10% der Distanz nicht unterschritten werden darf. Damit ergibt sich als minimal erlaubter Abstand 0,03 Meter. Der maximale Qualitätsverlust wird damit vor einer Kollision mit der Randbebauung erreicht. Als maximalen Qualitätsverlust wird der Wert 100 definiert.

**Maximale Ablage im Fahrstreifen:** Dieses Bewertungskriterium beschreibt die maximale Ablage im Fahrstreifen, während der Fahrt durch die Verschwenkung. Die Fahrstreifenbreite wird durch den lateralen Abstand der Randbebauungselemente beschrieben, die die linke und rechte Seite der Arbeitsstelle begrenzen. Die Ablage wird durch den Abstand zwischen dem Referenzpunkt des Fahrzeugs, welcher in der Mitte der Hinterachse liegt, und der Mitte des Fahrstreifens bestimmt. Als Qualitätsverlustfunktion wird die Zielwertfunktion gewählt, da sowohl positive als auch negative Abweichungen zu einem Qualitätsverlust führen. Der Zielwert wird als 0 Meter definiert, da das Fahrzeug idealerweise in der Mitte des eigenen Fahrstreifens fährt. Die erlaubte Abweichung ist die Hälfte der Differenz aus der Fahrstreifenbreite und der Fahrzeugbreite. Bei einer Überschreitung der erlaubten Abweichung verlässt das Fahrzeug den eigenen Fahrstreifen, was zu einer Kollision mit der Randbebauung führen könnte. Der maximale Qualitätsverlust wird auf 100 festgelegt. Damit wird das Bewertungskriterium gleichstark wie der minimale Abstand zur Randbebauung bewertet.

<sup>7</sup>Der Versuchsträger hat eine Breite von Spiegel von 1,85 Meter und mit Spiegeln von 2,09 Meter. In den Versuchen wurde die Breite des Fahrzeugs ohne Spiegel bemessen, da die Höhe der Leitwände beziehungsweise Leitkegel nicht bis zum Spiegel reichen. Für die Leitbake wurde die Breite des Fußes angenommen, welcher breiter ist als die Leitbake. Damit würde bei einer Kollision mit dem Fuß, die Bake den Spiegel nicht treffen.

Die Tabelle 12.5 fasst die Bewertungskriterien zusammen.

Bewertungskriterium	Zielwert $m$	erlaubte Abweichung $\Delta_0$	Qualitätsverlust $A_0$	Qualitätsverlustfunktion
min. Abstand zur Randbebauung	-	0,03 m	100	max
max. Ablage im Fahrstreifen	0 m	(Fahrstreifenbreite - Fahrzeugbreite)/2	100	Zielwert

**Tabelle 12.5:** Bewertungskriterien für die Auswertung der Tests.

### 12.5.2 Bewertung und Analyse der Testergebnisse

Im nächsten Schritt werden die erfassten Testergebnisse mittels der Testfallauswertung des Testbaukastens bewertet und analysiert. Mit Hilfe der Qualitätsverlustfunktionen wird der jeweilige Qualitätsverlust für die verschiedenen Bewertungskriterien berechnet. Zur Analyse wird die Mittelwertanalyse angewendet, um die Haupteffekte der einzelnen Parameterstufen auf die Bewertungskriterien zu schätzen. Hierfür werden die Qualitätsverluste der jeweiligen Parameterstufen gemittelt und daraus die Wirkungen der Parameterstufe geschätzt. Dabei ist zu beachten, dass derzeit keine Abhängigkeiten zwischen den Parametern und weiteren Informationsquellen modelliert wurden und dadurch gegebenenfalls aufgrund von technischen Restriktionen nicht zu bestehende Testfälle beziehungsweise Parameterkombinationen abgeleitet wurden. Diese werden im Folgenden dennoch analysiert, um die Grenzen der Fahrfunktion aufzuzeigen und zu analysieren unter welchen Randbedingungen ein Testfall bestanden oder nicht mehr bestanden wurde.

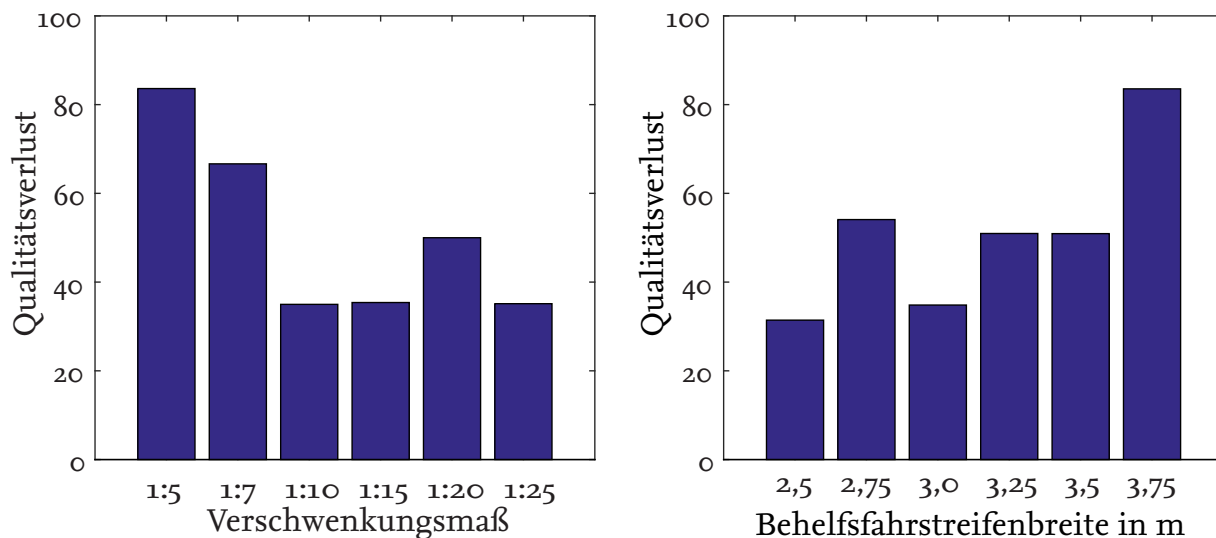
Insgesamt wurden 18 der 36 Testfälle bestanden. 18 Testfälle wurden nicht bestanden, da es zu einer Kollision mit der Randbebauung kam oder der eigene Fahrstreifen verlassen wurde. Durch die kombinatorische Testfallableitung ist jedes Paar von Diskretisierungsstufen in mindestens einem Testfall vorhanden. Des Weiteren wird durch die kombinatorische Testfallableitung jede Diskretisierungsstufe eines Einflussparameters gegenüber den übrigen Stufen eines Parameters im Mittel gleich oft getestet. Tabelle 12.6 zeigt die Häufigkeit der Diskretisierungsstufen innerhalb der erstellten Testsuite.

Im Folgenden wird der Haupteffekt von jeweils zwei Eingangsparametern auf die gewählten Bewertungskriterien durch die Testfallauswertung abgeschätzt. Dazu wurde zunächst der Qualitätsverlust für die Testfälle berechnet und über die Analyse der Mittelwerte der Einfluss auf das Bewertungskriterium geschätzt. Weiterhin wurde der Bravais-Pearson-Korrelationskoeffizient (Kohn, 2005, S.121f) berechnet, um den Zusammenhang der Ergebnisse zu schätzen. Die detaillierten Ergebnisse von jedem Testfall sind im Anhang der vorliegenden Arbeit zu finden.

Einflussparameter	Diskretisierungsstufen und deren Häufigkeit					
Verschwenkungsablage in m	3,5	7,0	10,5	14,0		
Häufigkeit	9	8	10	9		
Verschwenkungsmaß	1 : 5	1 : 7	1 : 10	1 : 15	1 : 20	1 : 25
Häufigkeit	6	6	6	6	6	6
Typ der Randbebauung	Leit- bake	Leit- kegel	Leit- wand			
Häufigkeit	12	12	12			
Abstand der Randbebauung in m	5,0	7,0	9,0	11,0	13,0	
Häufigkeit	7	7	7	7	8	
Behelfsfahrstreifenbreite in m	2,5	2,75	3,0	3,25	3,5	3,75
Häufigkeit	6	6	6	6	6	6

**Tabelle 12.6:** Häufigkeit der Diskretisierungsstufen in den kombinatorisch erstellten Testfällen

**Minimaler Abstand zur Randbebauung:** Die Ergebnisse der Analyse des Bewertungskriteriums minimaler Abstand zur Randbebauung sind in Abbildung 12.11 dargestellt. Es wurde der Einfluss des Verschwenkungsmaßes und der Behelfsfahrstreifenbreite auf das Bewertungskriterium analysiert.



**Abbildung 12.11:** Qualitätsverlust durch den minimalen Abstand zur Randbebauung bewertet anhand der Eingangsparameter Verschwenkungsmaß (links) und Behelfsfahrstreifenbreite (rechts)

Die Ergebnisse zeigen, dass zwischen dem Verschwenkungsmaß und dem Qualitätsverlust eine positive Korrelation ( $R = 0,8651$ ) vorhanden ist. Für die Berechnung des Korrelationskoeffizienten wurde der Kehrwert des Verschwenkungsmaßes verwendet, um eine korrekte Abbildung der Werte für die Berechnung zu erhalten. Daher ergibt sich ein Vorzeichenwechsel im Korrelationskoeffizienten. Damit berechnet sich, verglichen zum Verlauf in der Abbildung, trotz einer positiven Korrelation bei einem steigenden Verschwenkungsmaß ein fallender

Qualitätsverlust. Der Gesamtmittelwert des Qualitätsverlusts liegt bei 51. Eine signifikante Abweichung der Mittelwerte vom Gesamtmittelwert wurde nicht festgestellt.

Bei einem geringen Verschwenkungsmaß ergeben sich höhere Krümmungen für die Verschwenkung, sodass die Querführung ein größeres Lenkmoment aufbringen muss, um durch die Verschwenkungen zu fahren. Dabei zeigt sich, dass das Assistenzsystem dichter an der Randbebauung entlang fährt, als bei den Versuchen mit einem höheren Verschwenkungsmaß. Bei den Versuchen mit einem Verschwenkungsmaß von 1:5 und 1:7 zeigen sich die Grenzen der Fahrfunktion. Von den Versuchen auf der Stufe 1:5 wurde kein Testfall bestanden, da entweder der Fahrstreifen verlassen wurde oder es zu einer Kollision mit der Randbebauung kam. Dies liegt, wie oben schon beschrieben, an den hohen Krümmungen, die von der Fahrfunktion aufgrund von technischen Restriktionen nicht bestanden werden konnten. Auch eine Kombination mit den übrigen Diskretisierungsstufen, wie größere Fahrstreifenbreiten oder eine höhere Verschwenkungsablage, konnte dies nicht kompensieren. Daher wurden mit den gewählten Szenarien die Grenzen der Fahrfunktion gezeigt. Ein ähnliches Verhalten zeigt sich bei den Versuchen mit einem Verschwenkungsmaß von 1:7. Bei Versuchen mit dieser Diskretisierungsstufe wurden 3 von 6 Testfälle bestanden. Aufgrund des höheren Verschwenkungsmaß ergeben sich im Verhältnis zum Verschwenkungsmaß 1:5 geringe Krümmungen, sodass die Hälfte der Versuche bestanden wurde. Unter der Randbedingung einer Verschwenkungsablage von 3,50 Meter und 7 Meter ergeben sich jedoch immer noch Krümmungen, die von der Fahrfunktion nicht bewältigt werden können.

Das Ergebnis entspricht den erwarteten Ergebnissen, aufgrund der sich ergebenden Krümmungen durch das Verschwenkungsmaß. Der Effekt wurde aus der reduzierten Anzahl von 36 Testfällen geschätzt, ohne alle kombinatorisch durchführbaren Testfälle durchzuführen. Dadurch wird für dieses Bewertungskriterium und den Einflussparameter des Verschwenkungsmaßes gezeigt, dass mittels der kombinatorischen Testfallableitung und der Mittelwertanalyse die Haupteffekte geschätzt werden können. An dieser Stelle ist zu kritisieren, dass aufgrund der fehlenden Bedingungen zwischen den Parametern und Diskretisierungsstufen auch Testfälle abgeleitet wurden, die nicht zu bestehen waren. Mit Hilfe der kombinatorischen Testfallableitung konnte dennoch gezeigt werden, unter welchen Bedingungen die Testfälle bestanden oder nicht mehr bestanden wurden.

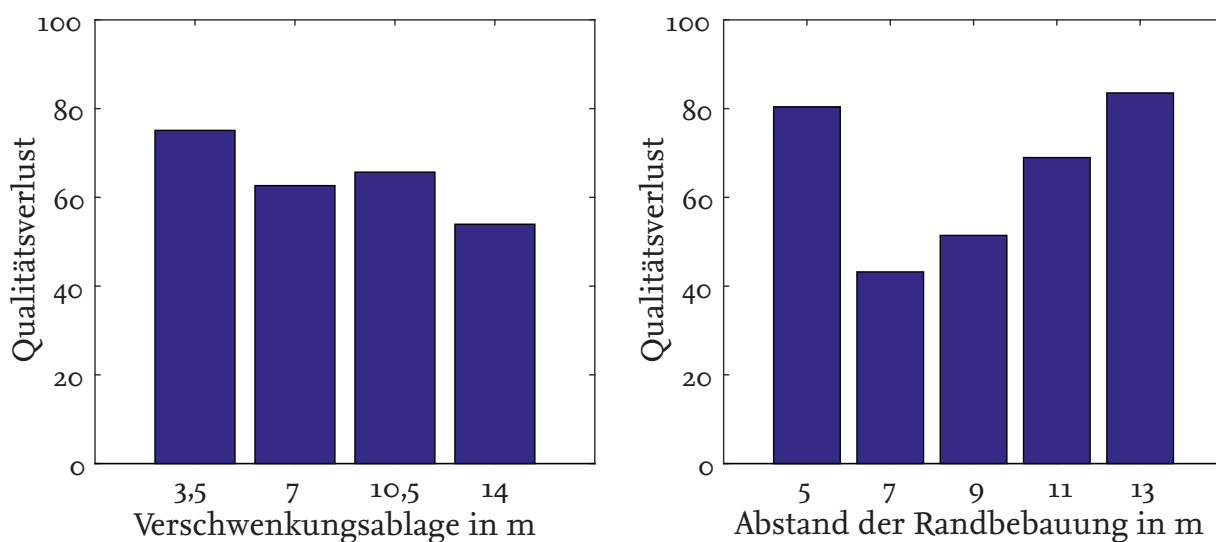
Weiterhin wurde der Einfluss der Behelfsfahrstreifenbreite auf das Bewertungskriterium geschätzt. Dabei wurde gezeigt, dass eine positive Korrelation ( $R = 0,4442$ ) zwischen der Behelfsfahrstreifenbreite und dem Qualitätsverlust vorhanden ist. Der Gesamtmittelwert des Qualitätsverlusts liegt bei 51. Eine signifikante Abweichung der Mittelwerte vom Gesamtmittelwert wurde nicht festgestellt.

Bei höheren Behelfsfahrstreifenbreiten ist ein geringerer Abstand zur Randbebauung festzustellen. Dieses Verhalten kann durch die Art der Umsetzung des Systems begründet werden. Da das System keine Fahrstreifenmittelführung realisiert, sondern nur reaktiv auf die Elemente der Arbeitsstelle reagiert, hat das Fahrzeug bei einer höheren Behelfsfahrstreifenbreite höhere Variationsbereiche sich im Fahrstreifen auszurichten. Dadurch kommt es vor, dass das Fahrzeug abwechselnd dicht an die linke und rechte Randbebauung der Verschwenkung fährt. Damit besteht die Möglichkeit, dass sich dieses Verhalten aufschwingt und das Fahrzeug in einem steileren Winkel auf die Randbebauung zufährt. In diesen Fällen müsste das System stärker eingreifen. Aufgrund eines begrenzten Lenkmoments benötigt das System jedoch länger um das Fahrzeug auszurichten, wodurch es dichter an die Randbebauung fährt.



Dieses Verhalten wird bei Verschwenkungen mit einer geringeren Behelfsfahrstreifenbreite verhindert, da dort das Fahrzeug durch die Assistenzfunktion und den geringen Querabstand zur Randbebauung stetig in der Mitte des Fahrstreifens geführt wird.

**Maximale Ablage im Fahrstreifen:** Das zweite Bewertungskriterium bewertet die maximale Fahrstreifenablage während der Fahrt durch die Verschwenkung. Es wird der Einfluss der Parameter der Verschwenkungsablage und der Abstand der Randbebauung auf dieses Bewertungskriterium geschätzt. Abbildung 12.12 zeigt die Ergebnisse der Analyse.



**Abbildung 12.12:** Qualitätsverlust durch die maximale Fahrstreifenablage bewertet anhand der Eingangsparameter Verschwenkungsablage (links) und Abstand der Randbebauung (rechts)

Zwischen dem Einfluss der Verschwenkungsablage und dem Qualitätsverlust durch die maximale Ablage im Fahrstreifen wird eine negative Korrelation ( $R = -0,6333$ ) festgestellt. Dadurch führt eine geringe Verschwenkungsablage zu einem hohen Qualitätsverlust. Der Gesamtmittelwert des Qualitätsverlusts liegt bei 66,8. Eine signifikante Abweichung der Mittelwerte vom Gesamtmittelwert wurde nicht festgestellt.

Dieses Ergebnis kann ebenfalls durch die hohen Krümmungen bei einer kleinen Verschwenkungsablage begründet werden. Die genauere Analyse der Testergebnisse zeigt, dass von den 9 Testfällen, die mit einer Verschwenkungsablage von 3,5 Meter durchgeführt wurden, 6 Testfälle nicht bestanden wurden. Die übrigen 3 Testfälle wurden erst durch die Kombination mit einem Verschwenkungsmaß von 1:20 beziehungsweise 1:25 oder mit einer größeren Fahrstreifenbreite von 3,5 Meter bestanden.

An dieser Stelle zeigt sich ebenfalls die schon beschriebene Schwäche durch die nicht modellierten Abhängigkeiten der Parameter und Diskretisierungsstufen. Durch die Verschwenkungsablage ergeben sich hohe Krümmungen, die von der Fahrfunktion unter den gegebenen technischen Randbedingungen nicht bestanden werden können. Allerdings zeigte eine genauere Analyse der Ergebnisse, dass bei einer Verschwenkungsablage von 3,5 Meter bei einem Verschwenkungsmaß von 1:20 beziehungsweise 1:25 die Tests bestanden wurden. Dadurch konnte dargestellt werden, unter welchen Randbedingungen die Tests bestanden und nicht bestanden wurden. An dieser Stelle sind weitere Forschungsarbeiten notwendig,

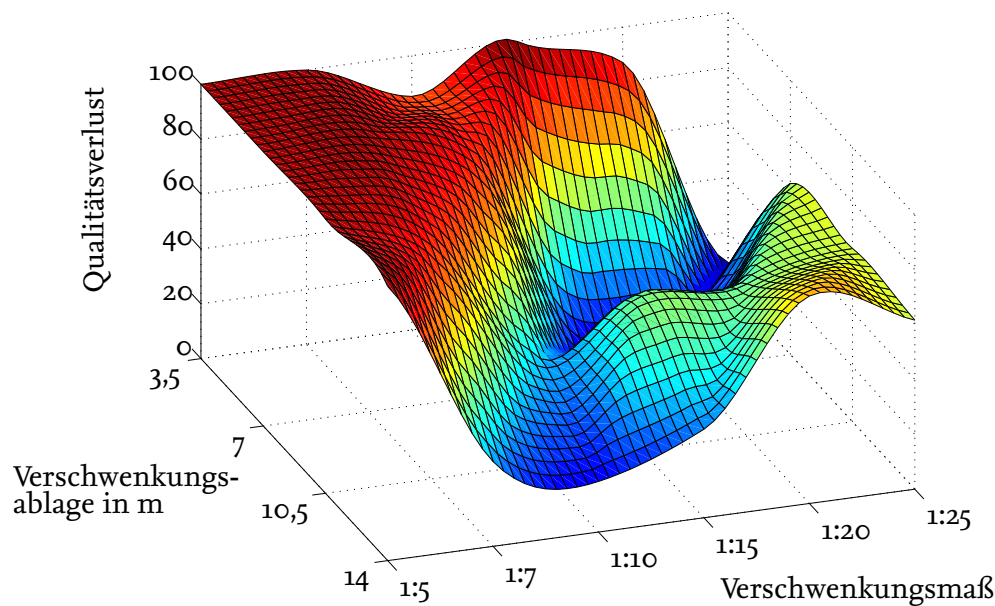
um die Abhängigkeiten zwischen Parametern und Diskretisierungsstufen aus verschiedensten Informationsquellen zu modellieren, die bei der Testfallableitung beachtet werden, um auf diese Weise die Ableitung von Testfällen zu vermeiden, die zum Beispiel aufgrund von technischen Restriktionen nicht zu bestehen sind.

Für den Abstand der Randbebauung wird für den Qualitätsverlust für die Stufen von 7 bis 13 Meter Abstand eine positive Korrelation ( $R = 0,7881$ ) berechnet. Damit ergibt sich bei einem hohen Abstand der Randbebauung ein höherer Qualitätsverlust. Auffällig ist die Diskretisierungsstufe von 5 Meter Abstand, da diese einen Qualitätsverlust aufweist, der der positiven Korrelation widerspricht. Bei der Beachtung dieses Werts bei der Berechnung der Korrelation ergibt sich ein Korrelationskoeffizient von  $R = -0,2454$ . Der Gesamtmittelwert des Qualitätsverlusts liegt bei 66,8. Eine signifikante Abweichung der Mittelwerte vom Gesamtmittelwert wurde nicht festgestellt.

Bei der genaueren Analyse der Testfälle, die mit dieser Diskretisierungsstufe durchgeführt wurden, zeigt sich, dass von den 7 Tests 3 Tests mit einer Verschwenkungsablage von 3,5 Meter durchgeführt wurden und somit der hohe Qualitätsverlust auf diese Diskretisierungsstufe zurückzuführen ist. Dieser Effekt zeigt noch einmal die oben beschriebene Schwäche durch die Nichtbeachtung von Nebenbedingungen. Weiterhin ist zu beachten, dass durch die starke Reduktion der Testfälle, welche durch die kombinatorische Ableitung hervorgerufen wird, die Möglichkeit besteht, dass der Einfluss von anderen Parametern den eigentlichen Effekt beeinflusst und eventuell dominiert, wie es bei der Diskretisierungsstufe von 5 Meter Abstand der Fall ist. Um diese Schwäche zu beheben, besteht die Möglichkeit die kombinatorische Testabdeckung zu erhöhen, beispielsweise auf eine *3-wise* (Übersetzung: 3-weise) Testabdeckung. Dadurch erhöht sich die Anzahl der Testfälle in der Testsuite, wodurch sich aber die Effekte besser abschätzen lassen. Für die gewählten Parameter und Diskretisierungsstufen müssten für eine *3-wise* (Übersetzung: 3-weise) Testabdeckung 184 Testfälle durchgeführt werden. Bei einer *4-wise* (Übersetzung: 4-weisen) Testabdeckung wären es 745 Testfälle. Die *5-wise* (Übersetzung: 5-weisen) Testabdeckung entspricht der Anzahl aller möglichen Kombinationen mit 2160 Testfällen.

**Analyse des Einflusses von zwei Parametern:** Abschließend wird der Einfluss von zwei Parametern auf die Bewertungskriterien des minimalen Abstands zur Randbebauung und der maximalen Ablage im Fahrstreifen bewertet. Dabei wird das Maximum des Qualitätsverlusts der beiden Bewertungskriterien in der Grafik auf der z-Achse aufgetragen. Die Bewertung findet anhand der Einflussparameter der Verschwenkungsablage und des Verschwenkungsmaßes statt. Abbildung 12.13 zeigt das Ergebnis der Analyse in Form einer 3D-Hüllfläche. Auf der z-Achse wird der maximale Mittelwert des Qualitätsverlusts aufgetragen. Die x-Achse beschreibt die Verschwenkungsablage und die y-Achse das Verschwenkungsmaß.

Das Ergebnis der Analyse zeigt, dass ein geringes Verschwenkungsmaß und eine geringe Verschwenkungsablage zu einem maximalen Qualitätsverlust führen, wodurch diese Testfälle nicht bestanden wurden. Der Qualitätsverlust der Tests mit einer Verschwenkungsablage von 3,5 Meter wird erst bei einem Verschwenkungsmaß von 1:20 reduziert. Weiterhin zeigt die Analyse, dass die Verschwenkung bei einer Verschwenkungsablage von 7 Meter bereits bei einem Verschwenkungsmaß von 1:10 durchfahren werden kann. Insgesamt wird festgestellt, dass mit steigender Verschwenkungsablage die Verschwenkung mit einem geringeren Verschwenkungsmaß befahrbar ist. Dies wird durch die größeren Kurvenradien begründet, die sich bei größeren Verschwenkungsablagen ergeben. Bei einer geringen Verschwenkungsablage



**Abbildung 12.13:** Maximaler Qualitätsverlust bei einer Durchfahrtsgeschwindigkeit von  $60 \frac{\text{km}}{\text{h}}$  durch den minimalen Abstand zur Randbebauung und der Ablage im eigenen Fahrstreifen anhand der Einflussparameter Verschwenkungsablage und Verschwenkungsmaß

wird ein höheres Verschwenkungsmaß benötigt, um die Kurvenradien auf gleiche Maße zu erhöhen. Damit kann eine sich verstärkende Wechselwirkung zwischen diesen beiden Einflussparametern geschätzt werden.

Die Ergebnisse der Auswertung zeigen ebenfalls die Nachteile dieses Verfahrens. Beispielsweise wurde bei einem Verschwenkungsmaß von 1:20 und einer Verschwenkungsablage von 14 Meter ein Qualitätsverlust von 80 berechnet. Dieser erhöhte Qualitätsverlust lässt sich nach einer detaillierten Analyse der einzelnen Testfälle auf einen hohen Abstand der Randbebauung zurückführen, welche im Testfall gegebenenfalls zu spät wahrgenommen wurde. Damit zeigt sich, dass zum Teil eine detailliertere Analyse durchzuführen ist, um die Ergebnisse zu interpretieren. Zudem wurden aufgrund von nicht modellierten Abhängigkeiten zwischen den Parametern auch nicht zu bestehende Testfälle abgeleitet. Mit Hilfe der kombinatorischen Testfallableitung konnte jedoch gezeigt werden, unter welchen Bedingungen Testfälle bestanden und nicht bestanden wurden.

## 12.6 Zusammenfassung

In diesem Kapitel wurde der modulare virtuelle Testbaukasten auf das Fallbeispiel des Engstellenassistenten angewendet. Dazu wurde zunächst die Fahrfunktion des Engstellenassistenten vorgestellt und anschließend die umgesetzten Module der funktionalen Systemarchitektur von Matthaei (2015) eingeordnet.

Anschließend wurde die Auswahl und Analyse von Einflussparametern durchgeführt. Dazu wurden verfügbaren Informationsquellen identifiziert. Für den Engstellenassistenten standen Normen und Richtlinien, Expertenwissen, Messfahrten als auch Fahrzeugkataloge als Informationsquellen zur Verfügung. Anhand dieser Quellen wurde das Anwendungsszenario einer Verschwenkung identifiziert. Die Einflussparameter zur Beschreibung der Verschwenkung

wurden anhand von Expertenwissen und geltenden Normen und Richtlinien bestimmt und diskretisiert.

Für die Testfallerstellung wurde das 4-Ebenen-Modell für die Beschreibung von Szenarien verwendet. Als Basisstrecke wurden Teile der Teststrecke der Volkswagen AG in Ehra-Lessien vermessen und für die Simulation rekonstruiert. Auf der zweiten Ebene des Modells wurden Arbeitsstellen in Form von verschiedenen Verschwenkungen definiert, die durch die identifizierten Einflussparameter beschrieben wurden. Auf der dritten Ebene wurde die Möglichkeit geschaffen, das dynamische Verhalten von Lastkraftwagen zu beschreiben, die ebenfalls durch die Verschwenkungen fahren. Die vierte Ebene des Modells wurde für die Tests des Engstellenassistenten nicht verwendet. Auf Basis der Szenarien wurden mittels der kombinatorischen Testfallableitung konkrete Testfälle erstellt. Dabei handelte es sich um Szenarien, bei denen keine beweglichen Elemente vorhanden waren. Für die Testfallableitung wurde ein kombinatorischer Algorithmus verwendet, um die Anzahl der Testfälle bei einer definierten Testabdeckung zu reduzieren. Als Testabdeckung wurde eine paarweise Abdeckung gewählt. Durch nicht modellierte Bedingungen zwischen den Parametern und Diskretisierungsstufen wurden auch aufgrund von technischen Restriktionen nicht zu bestehende Testfälle erstellt. Diese wurden verwendet, um die Grenzen der Fahrfunktion zu zeigen und auf diese Weise zu analysieren, unter welchen Bedingungen ein Testfall bestanden oder nicht mehr bestanden wurde.

Für die Testdurchführung wurde mittels der vorgestellten Zuordnungsmethode das *Vehicle-in-the-Loop* Verfahren aus den möglichen X-in-the-Loop-Verfahren für die Durchführung der Tests ausgewählt. Anschließend wurden die entwicklungsbegleitenden Testtreiber vorgestellt. Die Testtreiber wurden ebenfalls für die Durchführung der kombinatorisch erstellten Testfälle verwendet.

Mittels der Testfallauswertung wurden die Testergebnisse bewertet und analysiert. Dazu wurden zunächst Bewertungskriterien definiert. Für jedes Bewertungskriterium wurden eine Qualitätsverlustfunktion sowie ein Zielwert, die erlaubten Abweichungen vom Sollwert und der maximaler Qualitätsverlust für das jeweilige Bewertungskriterium definiert. Durch die Analyse der Mittelwerte wurden zum Teil die Haupteffekte der Einflussparameter geschätzt. Weiterhin konnte aufgrund der kombinatorischen Testfallableitung analysiert werden, unter welchen Bedingungen ein Testfall bestanden oder nicht mehr bestanden wurde. Die Analyse hat jedoch auch gezeigt, dass durch die stark reduzierte Anzahl von Testfällen die Haupt- und Wechselwirkungen nicht immer eindeutig zu zuordnen, wodurch genauere Analysen notwendig sind, die derzeit nicht automatisiert sind.



# Teil V

## Zusammenfassung und Ausblick





## 13 Zusammenfassung und Ausblick

In diesem abschließenden Kapitel werden die Inhalte der vorliegenden Arbeit zusammengefasst. Weiterhin werden der Beitrag dieser Arbeit zur Beantwortung der gestellten Forschungsfragen erläutert und ein Ausblick auf die weiteren Herausforderungen für die Absicherung und den Test von automatisierten Fahrfunktionen gegeben.

### 13.1 Zusammenfassung

Bei der bisherigen Entwicklung von automatisierten Fahrfunktionen lag der Schwerpunkt auf der funktionalen Entwicklung der Fahrfunktionen. Die Realisierung der Fahrfunktionen wurde in zahlreichen Demonstrationen von verschiedenen Fahrzeugherstellern und Universitäten gezeigt. Testkonzepte für die Freigabe der Fahrfunktionen für den öffentlichen Straßenverkehr wurden mit den Demonstrationen nicht gezeigt. In der vorliegenden Arbeit lag der Fokus nicht auf der Entwicklung der Fahrfunktionen, sondern auf der Herleitung eines systematischen Testkonzepts für automatisierte Fahrfunktionen.

Im ersten Teil der vorliegenden Arbeit wurde die Notwendigkeit eines systematischen Testkonzepts für automatisierte Fahrfunktionen hergeleitet. Dazu wurde das Testdilemma automatisierter Fahrfunktionen dargestellt. Weiterhin wurde die heutige und mögliche zukünftige Rolle der Forschungsabteilungen von Automobilherstellern bezüglich des Testdilemmas präsentiert. Aufbauend auf dem Testdilemma wurde die These aufgestellt, dass ein szenariobasiertes Testkonzept eine mögliche Lösung für das Testdilemma darstellen könnte. Es wurden sechs Forschungsfragen aufgestellt und erläutert, die für die Entwicklung eines szenariobasierten Testkonzepts beantwortet werden müssen.

Im zweiten Teil wurden die Grundlagen und Voraussetzungen dieser Arbeit dargestellt. Dazu wurden Begriffe und Definitionen aus dem Bereich des Testens, zur Beschreibung von Schnittstellen und zur Beschreibung von Straßen vorgestellt. Weiterhin wurde der Begriff der Komplexität von Szenarien untersucht und Dimensionen für die Bewertung der Komplexität abgeleitet. Diese wurden verwendet, um einen Ansatz einer Metrik für die Komplexität von Szenarien zu präsentieren. Darauffolgend wurden die funktionalen Beschreibungen für Fahrfunktionen verschiedener Automatisierungsgrade und eine funktionale Systemarchitektur für automatisierte Fahrfunktionen vorgestellt. Dieser Teil schließt mit einer Vorstellung von Methoden des Softwaretestens und der Verkehrssimulation sowie einer Klassifikation von verschiedenen X-in-the-Loop-Verfahren ab.

Im dritten Teil wurde das Testkonzept des modularen virtuellen Testbaukastens vorgestellt. Dazu wurden generelle Anforderungen an das Testkonzept erläutert. Weiterhin wurden Anforderungen vorgestellt, die durch Anwender des Testkonzepts und durch die Testobjekte gestellt werden. Darauffolgend wurden der Aufbau und die Schnittstellen zwischen den Modulen des Testbaukastens präsentiert. Für die Module Auswahl und Analyse von Einflussparametern, Testfallerstellung und Testdurchführung mittels X-in-the-Loop-Verfahren wurden Anforderungen definiert und ein ausgewählter Stand der Technik vorgestellt. Des Weiteren wurde aus dem Stand der Technik der Forschungsbedarf für jedes Modul abgeleitet. Für die Auswahl und Analyse von Einflussparametern wurde ein Schema für die Beschreibung

von Einflussparametern hergeleitet und verschiedene Informationsquellen für die Auswahl und Analyse von Einflussparametern bewertet. Im Rahmen der Testfallerstellung wurde ein generisches 4-Ebenen-Modell zur Beschreibung von Szenarien vorgestellt. Die systematische Testfallableitung wurde mittels kombinatorischer Algorithmen realisiert. Für die Testdurchführung wurde eine Methode für die Zuordnung von Testfällen auf X-in-the-Loop-Verfahren vorgestellt. Zusätzlich wurden Testtreiber analysiert, die für den Test von verschiedenen Modulen der vorgestellten funktionalen Systemarchitektur benötigt werden. Darauf aufbauend wurden die derzeitigen Testtreiber des modularen virtuellen Testbaukastens beschrieben. Für das Modul der Testfallauswertung wurden ausgewählte Methoden aus dem Stand der Technik präsentiert. Ein Beitrag zur Forschung wurde im Bereich der Testfallauswertung nicht geleistet. Der Teil schließt mit einer Beschreibung der Limitationen des Testbaukastens ab.

Der vierte Teil zeigte die Anwendung des Testbaukastens im Fallbeispiel des Engstellenassistenten. Das Projekt wurde vorgestellt und die verschiedenen Module des Testbaukastens angewendet.

## 13.2 Identifizierter Forschungsbedarf

Der identifizierte Forschungsbedarf basiert auf der These, die in Kapitel 1.2 aufgestellt wurde:

### These

*Ein szenariobasiertes Testkonzept, welches gezielt validierte virtuelle Umgebungen als Testwerkzeug einsetzt, bietet das Potential, die Herausforderung der Absicherung von zukünftigen automatisierten Fahrfunktionen durch eine systematische Erhöhung der Testabdeckung zu lösen.*

Für die Bestätigung oder Widerlegung der These wurde der Forschungsbedarf in sechs Forschungsfragen formuliert. Im Folgenden wird für die ersten fünf Forschungsfragen analysiert, welcher Beitrag zur Beantwortung der Frage geleistet wurde. Die sechste Forschungsfrage wurde in der vorliegenden Arbeit nicht weiter betrachtet.

### Forschungsfrage 1

*Welchen Aufbau bzw. welche Bestandteile muss ein szenariobasiertes Testkonzept für automatisierte Fahrfunktionen haben?*

Der Testbaukasten beinhaltet die Module der Auswahl und Analyse von Einflussparametern, die Testfallerstellung, die Testdurchführung und die Testfallauswertung. Anhand des Fallbeispiels wurde gezeigt, dass mit diesen Modulen eine systematische Ableitung, Durchführung und Auswertung von Testfällen für eine assistierende Fahrfunktion möglich ist. Weiterhin wurde gezeigt, dass das Testkonzept effizient und bedingt effektiv ist. Durch den gezielten Einsatz der Simulation für die Testdurchführung wurde der Aufwand für die Tests reduziert, wodurch die Effizienz gegenüber einer Testdurchführung mit realen Arbeitsstellen gesteigert wurde. Mit der Auswertung der Fallstudie wurde gezeigt, dass trotz der Reduktion der Anzahl von Testfällen zum Teil die erwarteten Ergebnisse hervorgebracht wurden. Allerdings wurden keine Abhängigkeiten zwischen den Parametern und Diskretisierungsstufen aus verschiedensten Informationsquellen modelliert, wodurch auch Testfälle abgeleitet wurden, die für das Testobjekt zum Beispiel aufgrund von technischen Randbedingungen nicht zu

schaffen waren. An dieser Stelle sind weiterführende Forschungsarbeiten notwendig, um die Abhängigkeiten zwischen den Parametern und Diskretisierungsstufen aus verschiedenen Informationsquellen zu modellieren und darauf aufbauend Testfälle abzuleiten. Hierbei stellt unter anderem die gleichzeitige Beachtung von verschiedensten Informationsquellen zur Ableitung von Abhängigkeiten eine Herausforderung dar. Weiterhin stellt sich die Frage, auf welcher Basis oder Begründung Szenarien gezielt weggelassen werden dürfen. Diese offenen Fragestellungen zeigen, wie wichtig die Auswahl und Analyse und darauffolgende Ableitung der Testfälle für ein systematisches Testkonzept ist.

Weiterhin wurde eine Anwendung auf automatisierte Fahrfunktionen mit höheren Automatisierungsgraden bisher nur theoretisch in einem eingeschränkten Rahmen exemplarisch gezeigt. An dieser Stelle sind weitere Forschungsarbeiten notwendig, um die praktische Anwendbarkeit des Testbaukastens zu zeigen.

Der Aufbau des modularen virtuellen Testbaukastens leistet daher einen Beitrag zur Beantwortung der Forschungsfrage. Der Testbaukasten gibt allerdings nur die äußere Struktur des Testkonzepts und die Schnittstellen zwischen den Modulen vor. Weiterhin wurde durch die Fallstudie des Engstellenassistenten nur eine mögliche praktische Anwendung des Testbaukastens gezeigt. In weiteren Forschungsarbeiten sollten weitere Methoden für die verschiedenen Module des Testbaukastens entwickelt werden, um den Testbaukasten weiterzuentwickeln und so an verschiedene Testobjekte anpassbar zu machen.

### **Forschungsfrage 2**

*Wie können Einflussparameter für den Test von automatisierten Fahrfunktionen systematisch abgeleitet werden und welche Herausforderungen entstehen dabei?*

Um einen Beitrag zur Beantwortung dieser Forschungsfrage zu leisten, wurden in der vorliegenden Arbeit ein Schema zur Beschreibung von Einflussparametern vorgestellt und verschiedene Informationsquellen für die Auswahl und Analyse von Einflussparametern bewertet. Dabei wurde festgestellt, dass die untersuchten Informationsquellen einen Beitrag zur Auswahl und Analyse von Einflussparametern leisten. Jedoch bleiben einige Lücken bestehen. Ein Beispiel ist die Diskretisierung wertkontinuierlicher Parameter für die Testfallableitung und die daraus resultierende Fehlerwahrscheinlichkeit durch die Diskretisierung. Derzeit bietet keine der analysierten Informationsquellen einen Ansatz zur Lösung dieser Herausforderung.

Die Forschungsfrage wurde somit in der vorliegenden Arbeit nicht komplett beantwortet. Ein Beitrag zur Beantwortung der Forschungsfrage wurde durch die Vorstellung des Schemas zur Beschreibung von Einflussparametern und die Bewertung der Informationsquellen geleistet, wodurch die Möglichkeit geschaffen wurde, eine systematische Auswahl und Analyse von Einflussparametern vorzunehmen. Um die identifizierten Lücken zu schließen, sind weitere Forschungsarbeiten auf diesem Gebiet notwendig.

### **Forschungsfrage 3**

*Wie kann ein Beschreibungsformat für Szenarien für automatisierte Fahrfunktionen aussehen und wie können aus diesen Szenarien Testfälle abgeleitet werden? Mit welchen Methoden kann die Anzahl der Testfälle bei einer bekannten Testabdeckung reduziert werden?*

Zur Beantwortung dieser Forschungsfrage wurde das generische 4-Ebenen-Modell zur Beschreibung von Szenarien vorgestellt. Dieses wurde in verschiedenen Diskussionsrunden

in den Entwurf des neuen Standards *OpenScenario* zur Beschreibung von Szenarien integriert. Damit wurde für den ersten Teil der Forschungsfrage ein Beitrag geleistet. Um einen Beitrag zur Beantwortung des zweiten Teils der Forschungsfrage zu leisten, wurde die kombinatorische Testfallableitung im Fallbeispiel des Engstellenassistenten evaluiert. Dabei wurde gezeigt, dass trotz einer starken Reduktion der Anzahl von Testfällen beim Bewertungskriterium *minimaler Abstand zur Randbebauung* die zu erwarteten Haupteffekte beobachtbar waren. Beim Bewertungskriterium *maximale Ablage im Fahrstreifen* kam es zu Mischeffekten, sodass die kombinatorische Testfallableitung derzeit nur eine eingeschränkte Antwort für den zweiten Teil der Forschungsfrage gibt. Weiterhin wurden keine Abhängigkeiten zwischen den Parametern und Diskretisierungsstufen modelliert, wodurch bei der Ableitung auch nicht für das Testobjekt zu bestehende Testfälle erstellt wurden. Aufgrund der kombinatorischen Testfallableitung konnte allerdings gezeigt werden, unter welchen Bedingungen Testfälle bestanden oder nicht mehr bestanden wurden. Zudem wurden rein stationäre Szenarien, d.h. ohne bewegliche Elemente, betrachtet. Somit sind weitere Forschungsarbeiten auf diesem Gebiet notwendig, um die Forschungsfrage zu beantworten. Dies gilt insbesondere bei der Betrachtung von Abhängigkeiten zwischen verschiedenen Parametern und Diskretisierungsstufen sowie Szenarien, bei denen bewegliche Elemente vorhanden sind.

#### Forschungsfrage 4

*Kann die Simulation verwendet werden, um den Testbedarf an real durchgeführten Testkilometern zu reduzieren? Wie können dafür X-in-the-Loop-Verfahren eingesetzt werden? Welche Herausforderungen ergeben sich bei der Erstellung von verschiedenen Testtreibern für Elemente einer funktionalen Systemarchitektur von automatisierten Fahrfunktionen?*

Um einen Beitrag zur Beantwortung dieser Forschungsfrage zu leisten, wurde eine Methode für eine Zuordnung von Testfällen auf X-in-the-Loop-Verfahren vorgestellt. Durch die Zuordnung der Testfälle auf die X-in-the-Loop-Verfahren wurde die Möglichkeit geschaffen, eine effiziente Testdurchführung zu realisieren. Weiterhin wurde eine Bewertung erstellt, welche Module der funktionalen Systemarchitektur von Matthaei (2015) derzeit mit Hilfe des Testbaukastens testbar und bewertbar sind. Die Bewertung hat gezeigt, dass die Module mit Ausnahme der Lokalisierungs- und Umfeldsensoren sowie der Kommunikation mit Hilfe des Testbaukastens testbar sind, da generell die Eingangsdaten zur Verfügung gestellt werden können. Weiterhin wurde gezeigt, dass für verschiedene Module die genauen Eingangs- als auch Ausgangsdaten erstellbar sind, wodurch diese Module mit Hilfe des Testbaukastens bewertbar sind. Um die Module der funktionalen Systemarchitektur zu testen, wurden im Rahmen der vorliegenden Arbeit und durch studentische Arbeiten für den Testbaukasten verschiedene Testtreiber entwickelt. Diese Testtreiber wurden in den Kapiteln 9.4 und 12.4.3 vorgestellt und im Projekt des Engstellenassistenten eingesetzt. Damit konnte gezeigt werden, dass Simulationen beziehungsweise X-in-the-Loop-Verfahren beim entwicklungsbegleitenden Test von automatisierten Fahrfunktionen eingesetzt werden können.

Bei der Bewertung der Testbarkeit der Module wurde weiterhin festgestellt, dass momentan Metriken fehlen, um die Simulationsmodelle in ihrer Güte zu bewerten. Daher besteht im Augenblick die Möglichkeit, mit Hilfe der Simulation die generelle Funktionalität der Module zu testen. Eine Optimierung oder Parametrisierung der Module ist aufgrund der fehlenden Metrik nur eingeschränkt möglich, wodurch für die Parametrisierung weiterhin Realtests notwendig sind. Weiterhin kann durch die fehlenden Metriken ein Vergleich zwischen realen

und virtuellen Testfahrten nur bedingt durchgeführt werden. Daher sind weitere Forschungsarbeiten notwendig, um Metriken zu entwickeln, die die Güte von Simulationsmodellen bewerten. Dadurch wird die Möglichkeit geschaffen, reale und virtuelle Messfahrten zu vergleichen. Dies ist notwendig, um Aussagen zu treffen, welche Szenarien in Zukunft in der Simulation anstelle von Realfahrten durchführbar sind.

### **Forschungsfrage 5**

*Wie können die Testergebnisse anhand von verschiedenen Kriterien bewertet werden und wie können die Einflüsse der Eingangsparameter anhand der Testergebnisse abgeschätzt werden?*

Um einen Beitrag zu dieser Forschungsfrage zu leisten, wurden die Qualitätsverlustfunktionen und die Mittelwertanalyse vorgestellt. Die Methoden wurden in der vorliegenden Arbeit nicht weiterentwickelt, sodass kein Forschungsbeitrag zur Methodenentwicklung geleistet wurde. Jedoch wurden die Methoden erstmalig im Bereich des Testens von automatisierten Fahrfunktionen eingesetzt. Anhand der Fallstudie wurde gezeigt, dass die Möglichkeit besteht, die Qualitätsverlustfunktionen für die Normierung von verschiedenen Bewertungskriterien zu verwenden. Die Funktionen sind derzeit auf mikroskopischer Ebene, d.h. Datenebene, anwendbar. Eine Verwendung der Qualitätsverlustfunktionen auf mesoskopischer oder makroskopischer Ebene ist derzeit nicht möglich. Hierfür sind weitere Forschungsarbeiten notwendig. Die Mittelwertanalyse wurde im Fallbeispiel für die Analyse der Testergebnisse verwendet. Mit der Anwendung wurde gezeigt, dass die Möglichkeit besteht, die zu erwarteten Ergebnisse zu rekonstruieren und die Haupteffekte zu schätzen.

Durch die Verwendung der beschriebenen Methoden wurde ein Teil zur Beantwortung dieser Forschungsfrage beigetragen. Jedoch sind weitere Forschungsarbeiten notwendig, um die Methoden für detailliertere Analysen der Testergebnisse weiterzuentwickeln.

Das Fallbeispiel kann nach der Bewertung der Forschungsfragen zur eingeschränkten Validierung des Testbaukastens verwendet werden. In Kapitel 2.2.4 wurde der Begriff der Validierung mit einem Prozess zur Überprüfung, ob das entwickelte Konzept den Erwartungen und Bedürfnissen des Anwenders entspricht, beschrieben. Anhand des Fallbeispiels wurde gezeigt, dass eine systematische Ableitung, Durchführung und Auswertung von Testfällen mit Hilfe des modularen virtuellen Testbaukastens für den Engstellenassistenten möglich ist. Damit wurde das Testkonzept für den Engstellenassistenten mit der Einschränkung von stationären Szenarien validiert. Für eine allgemeine Validierung muss das Testkonzept mit Fahrfunktionen verschiedener Automatisierungsgrade verwendet werden. Dabei spielt besonders die Betrachtung von Szenarien mit beweglichen Elementen eine entscheidende Rolle. Die beweglichen Elemente wurden in der durchgeführten Fallstudie nicht betrachtet und sollten in fortführenden Forschungsarbeiten weiter verfolgt werden.

Durch die Analyse der Forschungsfragen und den geleisteten Forschungsbeitrag in der vorliegenden Arbeit konnte gezeigt werden, dass sich die aufgestellte These bewährt hat. Anhand der Fallstudie wurde gezeigt, dass ein szenariobasiertes Testkonzept die automatisierten Fahrfunktionen systematisch testet. Dabei bietet das vorgestellte szenariobasierte Testkonzept durch den Aufbau und die vorgestellten Module das Potential, die Testabdeckung systematisch zu erhöhen, da die Szenarien gezielt aus den Einflussparametern abgeleitet werden. Jedoch stellt die Erstellung von Szenarien und die Ableitung von Einflussparametern eine große Herausforderung des Ansatzes dar. Für die Testdurchführung mittels verschiedener X-in-the-Loop-Verfahren konnten individuelle Vor- und Nachteile identifiziert werden. Eine



weitere Herausforderung ist das Fehlen von Metriken zur Bewertung der Güte der Simulationsmodelle. Dadurch können die X-in-the-Loop-Verfahren nur eingeschränkt im Testkonzept verwendet werden. Somit ergibt sich weiterer Forschungsbedarf, um die aufgestellte These zu bestätigen. Der Forschungsbedarf wird im folgenden Abschnitt beschrieben.

### 13.3 Ausblick

Der in der vorliegenden Arbeit vorgestellte modulare virtuelle Testbaukasten stellt einen Beitrag für den systematischen Test von automatisierten Fahrfunktionen dar. Die Analyse der verschiedenen Module des Testbaukastens hat weitere offene Herausforderungen gezeigt.

Eine der offenen Herausforderungen ist die Auswahl und Analyse von Einflussparametern. Hierfür werden die Diskretisierung von wertkontinuierlichen Einflussparametern eine entscheidende Rolle bei der Beantwortung der offenen Forschungsfragen einnehmen.

Eine weitere Herausforderung besteht in der Ableitung von Testfällen aus den Szenarien. Durch die Vielzahl von möglichen Testfällen ist eine Reduktion der Testfälle notwendig, da nicht alle möglichen Testfälle durchgeführt werden können. Der in der vorliegenden Arbeit vorgestellte Ansatz der kombinatorischen Testfallableitung bietet das Potential, einen ersten Ansatz zur Reduktion von Testfällen darzustellen. Bei der Anwendung wurde jedoch deutlich, dass aufgrund von nicht modellierten Bedingungen zwischen den Parametern und Diskretisierungsstufen verschiedener Informationsquellen auch Testfälle abgeleitet wurden, die zum Beispiel aufgrund von technischen Randbedingungen nicht zu bestehen waren. An dieser Stelle sind weitere Forschungsarbeiten notwendig, um Ansätze zu finden, welche die Abhängigkeiten zwischen den Parametern modellieren, um diese bei der Testableitung zu betrachten. Zudem wurde der Ansatz bisher nur für stationäre Szenarien ohne bewegliche Elemente angewendet. Für die Integration von beweglichen Elementen in die kombinatorische Testfallableitung sind weitere Forschungsarbeiten notwendig.

Eine weitere Herausforderung ist die Definition von Metriken für Simulationsmodelle, um die Güte der Modelle zu bewerten. Damit wird die Möglichkeit geschaffen, eine Aussage zu treffen, welche Tests in der Simulation anstelle von Realtests durchführbar sind. Diese Herausforderungen wurde ebenfalls in Kapitel 9 diskutiert. Dort wurde gezeigt, dass besonders die Bewertung von simulierten Sensordaten eine Herausforderung für den Test in der Simulation darstellt. Mit entsprechenden Metriken könnten Aussagen über den Vergleich von simulierten und realen Sensordaten getroffen werden. Hierfür sind weitere Forschungsarbeiten notwendig, um Metriken zu entwickeln.

Ich glaube, dass die Simulation den Test von automatisierten Fahrfunktionen deutlich bereichert und das Potential hat, das Testdilemma der automatisierten Fahrfunktionen zu lösen. Es bleibt mir damit nur noch zu sagen: Es ist noch viel zu tun. Packen wir es an...

## Eigene Veröffentlichungen

- BERGMILLER, P.; SCHULDT, F.; MAURER, M.: Optimized control of an electric vehicle with functional actuator redundancy. In: *2012 IEEE International Conference on Vehicular Electronics and Safety (ICVES)*. Istanbul, Türkei : IEEE, 2012, S. 25–30
- BERGMILLER, P.; SCHULDT, F.; MAURER, M.: Reifenverschleißausgleich in Elektrofahrzeugen mit funktionaler Aktorredundanz. In: *13. VDI-Fachtagung Reifen - Fahrwerk - Fahrbahn*. Hannover, Deutschland, 2011, S. 333–336
- MATTHAEI, R.; RESCHKA, A.; BAGSCHIK, G.; ESCHER, M.; MENZEL, T.; RIEKEN, J.; SCHEIDE, T.; SCHULDT, F.; ULBRICH, S.; WENDLER, J. T.; HECKER, P.; MAURER, M.: Das Projekt Stadtpilot - Automatisiertes Fahren an der TU Braunschweig. In: *Zeitschrift für die gesamte Wertschöpfungskette Automobilwirtschaft* 01 (2015), Nr. 18, S. 12–23
- SCHOLZ, S.; CHLOSTA, S.; FRETER, S.; SCHULDT, F.: Prototypische Umsetzung eines Bau- und Engstellenassistenten. In: *AAET - Automatisierungssysteme, Assistenzsysteme und eingebettete Systeme für Transportmittel*. Braunschweig, Deutschland : ITS Niedersachsen, 2014, S. 291–314
- SCHULDT, F.; LICHTHE, B.; MAURER, M.; SCHOLZ, S.: Systematische Auswertung von Testfällen für Fahrfunktionen im modularen virtuellen Testbaukasten. In: *9. Workshop Fahrerassistenzsysteme*. Walting, Deutschland : Uni-DAS e.V., 2014, S. 169–179
- SCHULDT, F.; MENZEL, T.; MAURER, M.: Eine Methode für die Zuordnung von Testfällen für automatisierte Fahrfunktionen auf X-in-the-Loop Verfahren im modularen virtuellen Testbaukasten. In: *10. Workshop Fahrerassistenzsysteme*. Walting, Deutschland : Uni-DAS e.V., 2015, S. 171–182
- SCHULDT, F.; RESCHKA, A.; MAURER, M.: A Method for an Efficient Systematic Test Case Generation for Advanced Driver Assistance Systems in Virtual Environments. In: MAURER, M. (Hrsg.); WINNER, H. (Hrsg.); PROKOP, G. (Hrsg.): *Automotive Systems Engineering* Bd. 2. Springer-Verlag Berlin Heidelberg, 2016. – angenommen zur Veröffentlichung
- SCHULDT, F.; SAUST, F.; LICHTHE, B.; MAURER, M.; SCHOLZ, S.: Effiziente systematische Testgenerierung für Fahrerassistenzsysteme in virtuellen Umgebungen. In: *AAET - Automatisierungssysteme, Assistenzsysteme und eingebettete Systeme für Transportmittel*. Braunschweig, Deutschland : ITS Niedersachsen, 2013, S. 114 – 134
- ULBRICH, S.; MENZEL, T.; RESCHKA, A.; SCHULDT, F.; MAURER, M.: Defining and Substantiating the Terms Scene, Situation and Scenario for Automated Driving. In: *18th IEEE International Conference on Intelligent Transportation Systems (ITSC)*. Las Palmas, Spanien : IEEE, 2015, S. 982–988

- ULBRICH, S.; MENZEL, T.; RESCHKA, A.; SCHULDT, F.; MAURER, M.: Definition der Begriffe Szene , Situation und Szenario für das automatisierte Fahren. In: *10. Workshop Fahrerassistenzsysteme*. Walting, Deutschland : Uni-DAS e.V., 2015, S. 105–117
- ULBRICH, S.; SCHULDT, F.; MENZEL, T.; HOMEIER, K.; STEINHOFF, M.; KRAUSE, J.; MAURER, M.: Testing and Validating Tactical Lane Change Behavior Planning for Automated Driving. In: HORN, M. (Hrsg.); WATZENIG, D. (Hrsg.): *Automated Driving* Bd. 1. Springer International Publishing AG, 2016, S. 451–471
- ZUG, S.; STEUP, C.; SCHOLLE, J. B.; BERGER, C.; LANDSIEDEL, O.; SCHULDT, F.; RIEKEN, J.; MATTHAEI, R.; FORM, T.: Technical evaluation of the Carolo-Cup 2014 - A competition for self-driving miniature cars. In: *2014 IEEE International Symposium on Robotic and Sensors Environments (ROSE)*. Timisoara, Rumänien : IEEE, 2014, S. 100–105

## Betreute studentische Arbeiten

- BACHMANN, T.: *Entwurf einer Toolkette zur automatisierten Generierung von detaillierten Straßenmodellierungen in VTD auf Basis von Kartendaten*, Technische Universität Braunschweig, Bachelorarbeit, 2012
- KONRAD, J.: *Konfiguration eines statischen Fahrsimulators zur Durchführung von Driver-in-the-Loop Simulationen für den Test von Fahrerassistenzsystemen*, Technische Universität Braunschweig, Bachelorarbeit, 2014
- LACKMANN, M.: *Entwicklung und Implementierung eines Frameworks zur Nachbildung prinzipbedingter Messfehler von Sensordaten in der Simulation*, Technische Universität Braunschweig, Masterthesis, 2014
- MAREK, G.: *Entwicklung und Implementierung eines Frameworks zur systematischen Gestaltung und Variation von virtuellen Umgebungen in einer Fahrsimulation zur Testung von Fahrerassistenzsystemen*, Technische Universität Braunschweig, Masterthesis, 2014
- SEMRAU, M.: *Entwicklung eines Verfahrens zur Identifikation von ausgewählten Verkehrssituationen aus dem chinesischen Großstadtverkehr zur Bewertung von Fahrerassistenzsystemen*, Technische Universität Braunschweig, Masterthesis, 2014



# Literaturverzeichnis

- [Ammann und Offutt 1994] AMMANN, P.; OFFUTT, J.: Using formal methods to derive test frames in category-partition testing. In: *Ninth Annual Conference on Computer Assurance (COMPASS)*. Gaithersburg, MD, USA : IEEE, 1994, S. 69–79
- [Ammann und Offutt 2008] AMMANN, P.; OFFUTT, J.: *Introduction to Software Testing*. Bd. 1. Cambridge : Cambridge University Press, 2008
- [Arnicane 2009] ARNICANE, V.: Complexity of Equivalence Class and Boundary Value Testing Methods. In: *Computer Science and Information Technologies* 751 (2009), S. 80–101
- [Baake u. a. 2014] BAAKE, U.; WÜST, K.; MAURER, M.; LUTZ, A.: Testing and simulation-based validation of ESP systems for vans. In: *ATZ Worldwide* 116 (2014), Nr. 2, S. 30–35
- [Bagschik u. a. 2015a] BAGSCHIK, G.; CERNAT, H.; DIERKES, F.; MAURER, M.; MATTHAEI, R.; MENZEL, T.; NOLTE, M.; RESCHKA, A.; RIEKEN, J.; SCHULDT, F.; STOLTE, T.; ULBRICH, S.; WENDLER, J. T.: *Oberseminar: Chancen und Risiken für X-in-the-Loop Verfahren*, Technische Universität Braunschweig. 2015
- [Bagschik u. a. 2015b] BAGSCHIK, G.; CERNAT, H.; DIERKES, F.; MAURER, M.; MATTHAEI, R.; MENZEL, T.; NOLTE, M.; RESCHKA, A.; RIEKEN, J.; SCHULDT, F.; STOLTE, T.; ULBRICH, S.; WENDLER, J. T.: *Oberseminar: Wie müsste ein Prüfgelände für Fahrerassistenzsysteme gestaltet sein?*, Technische Universität Braunschweig. 2015
- [Bagschik u. a. 2015c] BAGSCHIK, G.; CERNAT, H.; DIERKES, F.; MAURER, M.; MENZEL, T.; NOLTE, M.; RESCHKA, A.; RIEKEN, J.; SCHULDT, F.; STOLTE, T.; ULBRICH, S.; WENDLER, J. T.: *Diskussionsrunden der Arbeitsgruppe Elektronische Fahrzeugsysteme: Detaillierung funktionaler Systemarchitektur*, Technische Universität Braunschweig. 2015
- [Bakaba u. a. 2012] BAKABA, J. E.; ENKE, M.; HEINE, A.; LIPPOLD, C.; MAIER, R.; ORTLEPP, J.; SCHULZ, R.: Untersuchung der Verkehrssicherheit im Bereich von Baustellen auf Bundesautobahnen / Gesamtverband der Deutschen Versicherungswirtschaft e. V. Berlin, 2012. – Forschungsbericht
- [Bartels u. a. 2015] BARTELS, A.; ROHLFS, M.; HAMEL, S.; SAUST, F.; KLAUSKE, L. K.: Querführungsassistentz. In: WINNER, H. (Hrsg.); HAKULI, S. (Hrsg.); LOTZ, F. (Hrsg.); SINGER, C. (Hrsg.): *Handbuch Fahrerassistenzsysteme: Grundlagen, Komponenten und Systeme für aktive Sicherheit und Komfort* Bd. 3. Wiesbaden : Springer Vieweg, 2015, S. 937–957



- [Bartels 2012] BARTELS, A.: Systembeschreibung automatischer Fahrfunktionen. In: GAS-SER, T. M. (Hrsg.); ARZT, C. (Hrsg.); AYOUBI, M. (Hrsg.); BARTELS, A. (Hrsg.); BÜRKLE, L. (Hrsg.); EIER, J. (Hrsg.); FLEMISCH, F. (Hrsg.); HÄCKER, D. (Hrsg.); HESSE, T. (Hrsg.); HUBER, W. (Hrsg.); LOTZ, C. (Hrsg.); MAURER, M. (Hrsg.); RUTH-SCHUMACHER, S. (Hrsg.); SCHWARZ, J. (Hrsg.); VOGT, W. (Hrsg.): *Rechtsfolgen zunehmender Fahrzeugautomatisierung: gemeinsamer Schlussbericht der Projektgruppe*. F83. 2012, S. 23–44
- [Bengler u. a. 2014] BENGLER, K.; DIETMAYER, K.; FÄRBER, B.; MAURER, M.; STILLER, C.; WINNER, H.: Three Decades of Driver Assistance Systems: Review and Future Perspectives. In: *IEEE Intelligent Transportation Systems Magazine* 6 (2014), Nr. 4, S. 6–22
- [Benmimoun u. a. 2012a] BENMIMOUN, M.; PÜTZ, A.; AUST, M. L.; FABER, F.; SÁNCHEZ, D.; METZ, B.; SAINT PIERRE, G.; GEISSLER, T.; GUIDOTTI, L.; MALTA, L.: *euroFOT SP6 D6.1 Final evaluation results*. 2012
- [Benmimoun u. a. 2012b] BENMIMOUN, M.; PÜTZ, A.; ZLOCKI, A.; ECKSTEIN, L.: Effects of ACC and FCW on Speed, Fuel Consumption, and Driving Safety. In: *2012 IEEE Vehicular Technology Conference (VTC Fall)*. Quebec City, QC, Kanada : IEEE, 2012, S. 1–6
- [Benmimoun u. a. 2010] BENMIMOUN, M.; FAHRENKROG, F.; BENMIMOUN, A.: Automatisierte Situationserkennung zur Bewertung des Potentials von Fahrerassistenzsystemen im Rahmen des Feldversuchs euroFOT. In: *26. VDI/VW-Gemeinschaftstagung - Fahrerassistenz und Integrierte Sicherheit* Bd. 26. Wolfsburg : VDI Verlag GmbH, 2010, S. 231–245
- [Benmimoun u. a. 2012c] BENMIMOUN, M.; FAHRENKROG, F.; ZLOCKI, A.; ECKSTEIN, L.: Erkennung und Klassifizierung Kritischer Fahrsituationen Mittels Fahrzeugdaten. In: *ATZ - Automobiltechnische Zeitschrift* 114 (2012), Nr. 10, S. 820–826
- [Berger 2010] BERGER, C.: *Automating Acceptance Tests for Sensor and Actuator-based Systems on the Example of Autonomous Vehicles*, Technische Universität Braunschweig, Dissertation, 2010
- [Berger u. a. 2014a] BERGER, C.; BLOCK, D.; HEEREN, S.; HONS, C.; KÜHNEL, S.; LESCHKE, A.; PLOTNIKOV, D.; RUMPE, B.: Simulations on Consumer Tests : A Systematic Evaluation Approach in an Industrial Case Study. In: *17th IEEE International Conference on Intelligent Transportation Systems (ITSC)*. Qingdao, China : IEEE, 2014, S. 1474–1480
- [Berger u. a. 2014b] BERGER, C.; BLOCK, D.; HEEREN, S.; HONS, C.; KÜHNEL, S.; LESCHKE, A.; PLOTNIKOV, D.; RUMPE, B.: Simulations on Consumer Tests: Systematic Evaluation of Tolerance Ranges by Model-Based Generation of Simulation Scenarios. In: *30. VDI/VW-Gemeinschaftstagung - Fahrerassistenz und Integrierte Sicherheit* Bd. 30. Wolfsburg, Deutschland : VDI Verlag GmbH, 2014, S. 1474–1480
- [Bergmiller 2014] BERGMILLER, P.: *Towards Functional Safety in Drive-by-Wire Vehicles*, Technische Universität Braunschweig, Dissertation, 2014

- [Bergmiller 2016] BERGMILLER, P.: *Persönliche Kommunikation: Simulation als Freigabewerkzeug*. 2016
- [Beyer 2002] BEYER, D.: *Formale Verifikation von Realzeit-Systemen mittels Cottbus Timed Automata*, Brandenburgische Technische Universität Cottbus-Senftenberg, Dissertation, 2002
- [Block u. a. 2014] BLOCK, D.; HEEREN, S.; KÜHNEL, S.; LESCHKE, A.; RUMPE, B.; SEREBRO, V.: Simulations on Consumer Tests: A Perspective for Driver Assistance Systems. In: *International Workshop on Engineering Simulations for Cyber-Physical Systems (ES4CPS '14)*. Dresden, Deutschland : ACM, 2014, S. 38–43
- [BMVI 2009] BUNDESMINISTERIUM FÜR VERKEHR: *Richtlinien für die Sicherung von Arbeitsstellen an Straßen*. Bd. 4. Bonn : Kirschbaum Verlag GmbH, 2009
- [BMW 2016] BMW: *Bedienungsanleitung BMW 7er*. 2016
- [Bock 2008] BOCK, T.: *Vehicle in the Loop*, Technische Universität München, Dissertation, 2008
- [Bock u. a. 2007] BOCK, T.; MAURER, M.; FÄRBER, G.: Validation of the Vehicle in the Loop (VIL); A milestone for the Simulation of Driver Assistance Systems. In: *2007 IEEE Intelligent Vehicles Symposium (IV)*. Istanbul, Türkei : IEEE, 2007, S. 612–617
- [Box und Draper 2007] BOX, G. P.; DRAPER, N. R.: *Response Surfaces, Mixtures, and Ridge Analyses*. Bd. 2. New Jersey : John Wiley & Sons, Inc, 2007
- [Brückner 2011] BRÜCKNER, J.: *Persönliche Kommunikation: Aufbau von Arbeitsstellen, ASS Verkehrsservice GmbH*. 2011
- [Broggi u. a. 2015] BROGGI, A.; CERRI, P.; DEBATTISTI, S.; LAGHI, M. C.; MEDICI, P.; MOLINARI, D.; PANCIOLO, M.; PRIOLETTI, A.: PROUD - Public Road Urban Driverless-Car Test. In: *IEEE Transactions on Intelligent Transportation Systems* 16 (2015), Nr. 6, S. 3508–3519
- [Broggi u. a. 2013] BROGGI, A.; BUZZONI, M.; DEBATTISTI, S.; GRISLERI, P.; LAGHI, M. C.; MEDICI, P.; VERSARI, P.: Extensive tests of autonomous driving technologies. In: *IEEE Transactions on Intelligent Transportation Systems* 14 (2013), Nr. 3, S. 1403–1415
- [Bubb u. a. 2015] BUBB, H.; VOLLRATH, M.; REINPRECHT, K.; MAYER, E.; KÖRBER, M.: Der Mensch als Fahrer. In: BUBB, H. (Hrsg.); BENGLER, K. (Hrsg.); GRÜNEN, R. E. (Hrsg.); VOLLRATH, M. (Hrsg.): *Automobilergonomie* Bd. 1. Wiesbaden : Springer Fachmedien, 2015, S. 67–162
- [Burdessa u. a. 2010] BURDESSA, C.; KREIS, C.; GREUL, R.; BERTRAM, T.: Eine Kamera zur Spurerkennung in einer virtuellen Umgebung. In: *15. VDI/VW-Gemeinschaftstagung - Erprobung und Simulation in der Fahrzeugentwicklung - Mess- und Versuchstechnik* Bd. 15. Düsseldorf : VDI Verlag GmbH, 2010, S. 311–322

- [Cao u. a. 2015] CAO, P.; WACHENFELD, W.; WINNER, H.: Perception sensor modeling for virtual validation of automated driving. In: *it - Information Technology* 57 (2015), Nr. 4, S. 243–251
- [Damböck 2013] DAMBÖCK, D.: *Automationseffekte im Fahrzeug – von der Reaktion zur Übernahme*, Technische Universität München, Dissertation, 2013
- [Dickmanns 2007] DICKMANN, E. D.: *Dynamic Vision for Perception and Control of Motion*. London : Springer-Verlag, 2007
- [Dijkstra 1972] DIJKSTRA, E. W.: The Humble Programmer. In: *Communications of the ACM* 15 (1972), Nr. 10, S. 859–866
- [DIN 5725 2002] DEUTSCHES INSTITUT FÜR NORMUNG (DIN): *DIN ISO 5725-2:2002-12: Genauigkeit (Richtigkeit und Präzision) von Messverfahren und Messergebnissen - Teil 2: Grundlegende Methode für Ermittlung der Wiederhol- und Vergleichpräzision eines vereinheitlichten Messverfahrens*. 2002
- [DMV 2014] DEPARTMENT OF MOTOR VEHICLES: *Autonomous Vehicles Testing Regulations*. 2014
- [DMV 2016] DEPARTMENT OF MOTOR VEHICLES: *Unfallbericht Google*. 2016. – URL [https://www.dmv.ca.gov/portal/wcm/connect/30fbc34a-2345-46f0-bdb2-ca1bac08f3a2/Google\\_042816.pdf?MOD=AJPERES](https://www.dmv.ca.gov/portal/wcm/connect/30fbc34a-2345-46f0-bdb2-ca1bac08f3a2/Google_042816.pdf?MOD=AJPERES). – Zugriffsdatum: 18.05.2016
- [Donges 1982] DONGES, E.: Aspekte der aktiven Sicherheit bei der Führung von Personenkraftwagen. In: *Automobil-Industrie* 27 (1982), Nr. 2, S. 183–190
- [Donges 1999] DONGES, E.: A Conceptual Framework for Active Safety in Road Traffic. In: *Vehicle System Dynamics* 32 (1999), Nr. 2-3, S. 113–128
- [Du u. a. 2004] DU, J.; MASTERS, J.; BARTH, M.: Lane-level positioning for in-vehicle navigation and automated vehicle location (AVL) systems. In: *7th International IEEE Conference on Intelligent Transportation Systems (ITSC)*. Washington, WA, USA : IEEE, 2004, S. 35–40
- [Dupuis u. a. 2010a] DUPUIS, M.; NEUMANN-COSEL, K. V.; WEISS, C.: Virtual Test Drive: Vereinheitlichung der Simulationsumgebung für SiL-, HiL-, DiL- und ViL-Tests bei der Entwicklung von Fahrerassistenz- und aktiven Sicherheitssystemen. In: *Autotest*. Stuttgart, Deutschland, 2010
- [Dupuis u. a. 2010b] DUPUIS, M.; STROBL, M.; GREZLIKOWSKI, H.: OpenDRIVE 2010 and Beyond – Status and Future of the de facto Standard for the Description of Road Networks. In: *Driving Simulation Conference DSC Europe*. Paris, Frankreich, 2010, S. 231–242
- [Eckstein 2014] ECKSTEIN, P. P.: *Statistik für Wirtschaftswissenschaftler*. 4. Wiesbaden : Springer Fachmedien Wiesbaden, 2014

- [Eisert 2015] EISERT, R.: US-Autozulieferer Delphi testet autonomes Fahren in Wuppertal. In: *Wirtschaftswoche*. 2015
- [Eltaher 2013] ELTAHER, A.: *Human-Like Test Systems: A Cognitive-Oriented Approach Applied to Infotainment Devices*, Technische Universität Braunschweig, Dissertation, 2013
- [Ernst 2015] ERNST, S.: *Konzeption und Implementierung von Algorithmen zur Manöverbestimmung wahrgenommener mehrspuriger Verkehrsteilnehmer für automatisierte Straßenfahrzeuge*, Technische Universität Braunschweig, Masterthesis, 2015
- [ERTICO-ITS Europe 2009] ERTICO-ITS EUROPE: *Response 3: Code of Practice for Design and Evaluation of ADAS*. 2009
- [Fecher u. a. 2015] FECHER, N.; HOFFMANN, J.; WINNER, H.: EVITA – Das Prüfverfahren zur Beurteilung von Antikollisionssystemen. In: WINNER, H. (Hrsg.); HAKULI, S. (Hrsg.); LOTZ, F. (Hrsg.); SINGER, C. (Hrsg.): *Handbuch Fahrerassistenzsysteme* Bd. 3. Wiesbaden : Springer Vieweg, 2015, S. 197–206
- [Feess 2015] FEESS, E.: *Gabler Wirtschaftslexikon, Stichwort: Komplexität*. 2015. – URL <http://wirtschaftslexikon.gabler.de/Archiv/5074/komplexitaet-v8.html>. – Zugriffsdatum: 21.11.2015
- [Feess u. a. 2015] FEESS, E.; LACKES, R.; SIEPERMANN, M.; STEVEN, M.; THOMMEN, J.; KAMPS, U.: *Gabler Wirtschaftslexikon, Stichwort: Effizienz*. 2015. – URL <http://wirtschaftslexikon.gabler.de/Archiv/7640/effizienz-v16.html>. – Zugriffsdatum: 21.11.2015
- [FGSV 1980] FORSCHUNGSGESELLSCHAFT FÜR STRASSEN UND VERKEHRSWESEN: *Richtlinie für die Markierung von Straßen*. Köln : Kirschbaum Verlag GmbH, 1980
- [FGSV 2006] FORSCHUNGSGESELLSCHAFT FÜR STRASSEN UND VERKEHRSWESEN: *Richtlinien für die Anlage von Stadtstraßen*. Köln : Kirschbaum Verlag GmbH, 2006
- [FGSV 2008] FORSCHUNGSGESELLSCHAFT FÜR STRASSEN UND VERKEHRSWESEN: *Richtlinien für die Anlage von Autobahnen*. Köln : Kirschbaum Verlag GmbH, 2008
- [FGSV 2013] FORSCHUNGSGESELLSCHAFT FÜR STRASSEN UND VERKEHRSWESEN: *Richtlinie für die Anlage von Landstraßen*. Köln : Kirschbaum Verlag GmbH, 2013
- [Fleischmann 2008] FLEISCHMANN, A.: *Modellbasierte Formalisierung von Anforderungen für eingebettete Systeme im Automotive-Bereich*, Technische Universität München, Dissertation, 2008
- [Form und Eltaher 2011] FORM, T.; ELTAHER, A.: *Persönliche Kommunikation: Expertwissen beim Testen*. 2011
- [Funk 2010] FUNK, S.: Emulation. In: NEUROTH, H. (Hrsg.); OSSWALD, A. (Hrsg.); SCHEFFEL, R. (Hrsg.); STRATHMANN, S. (Hrsg.); HUTH, K. (Hrsg.): *nestor Handbuch: Eine kleine Enzyklopädie der digitalen Langzeitarchivierung*. V2.3. Göttingen : nestor – Network of Expertise in Long-Term Storage of Digital Resources, 2010, S. 175–182

- [Gasser u. a. 2012] GASSER, T. M.; ARZT, C.; AYOUBI, M.; BARTELS, A.; BÜRKLE, L.; EIER, J.; FLEMISCH, F.; HÄCKER, D.; HESSE, T.; HUBER, W.; LOTZ, C.; MAURER, M.; RUTH-SCHUMACHER, S.; SCHWARZ, J.; VOGT, W.: Rechtsfolgen zunehmender Fahrzeugautomatisierung: gemeinsamer Schlussbericht der Projektgruppe. Bremerhaven : Wirtschaftsverlag NW, Verlag für neue Wissenschaft, 2012. – Forschungsbericht
- [Gerson u. a. 2011] GERSON, S.; JÄNSCH, D.; NIETSCHKE, W.; CANDERS, R.; CORNELSEN, K.; MAURER, M.; SCHUMACHER, W.; MEYER, H.: InDrive Simulator. In: *8th Symposium Hybrid- und Electric Vehicles*. Braunschweig, Deutschland : ITS Niedersachsen, 2011, S. 166–186
- [Gipps 1986] GIPPS, P.: A model for the structure of lane-changing decisions. In: *Transportation Research Part B: Methodological* 20 (1986), Nr. 5, S. 403–414
- [Gotzig und Geduld 2015] GOTZIG, H.; GEDULD, G.: LIDAR-Sensorik. In: WINNER, H. (Hrsg.); HAKULI, S. (Hrsg.); LOTZ, F. (Hrsg.); SINGER, C. (Hrsg.): *Handbuch Fahrerassistenzsysteme* Bd. 3. Wiesbaden : Springer Vieweg, 2015, S. 317–334
- [Grindal u. a. 2005] GRINDAL, M.; OFFUTT, J.; ANDLER, S. F.: Combination testing strategies: a survey. In: *Software Testing, Verification and Reliability* 15 (2005), Nr. 3, S. 167–199
- [Grochtmann und Grimm 1993] GROCHTMANN, M.; GRIMM, K.: Classification trees for partition testing. In: *Software Testing, Verification and Reliability* 3 (1993), Nr. 2, S. 63–82
- [Hamburg 2015] HAMBURG, M.: *ISTQB / GTB Standardglossar der Testbegriffe*. 2015
- [Hartmann 2001] HARTMANN, N.: *Automation des Tests eingebetteter Systeme am Beispiel der Kraftfahrzeugelektronik*, Universität Fredericiana Karlsruhe, Dissertation, 2001
- [Hendriks u. a. 2010] HENDRIKS, F.; PELDERS, R.; TIDEMAN, M.: Future Testing of Active Safety Systems. In: *SAE International Journal of Passenger Cars - Electronic and Electrical Systems* 3 (2010), Nr. 2, S. 170–175
- [Hilf u. a. 2010] HILF, K.; MATHEIS, I.; MAUSS, J.; RAUH, J.: Automated Simulation of Scenarios to Guide the Development of a Crosswind Stabilization Function. In: *6th IFAC Symposium on Advances in Automotive Control*. München, Deutschland, 2010, S. 768–772
- [Hoffmann 2008] HOFFMANN, D. W.: *Software-Qualität*. Bd. 1. Berlin, Heidelberg : Springer Vieweg, 2008
- [Holzmann u. a. 2012] HOLZMANN, H.; HAHN, K. M.; WEBB, J.; MIES, O.: Simulation-Based ESC Homologation for Passenger Cars. In: *ATZ Worldwide* 114 (2012), Nr. 9, S. 40–43
- [Homès 2012] HOMÈS, B.: *Fundamentals of Software Testing*. London, New York : ISTE Ltd, John Wiley & Sons, Inc., 2012
- [Howden 1978] HOWDEN, W. E.: An evaluation of the effectiveness of symbolic testing. In: *Software: Practice and Experience* 8 (1978), Nr. 4, S. 381–397



- [IEEE 610 1990] IEEE Standard Glossary of Software Engineering Terminology. IEEE, 1990. – Forschungsbericht
- [ISO 11270 2014] INTERNATIONAL ORGANIZATION FOR STANDARDIZATION (ISO): *ISO11270:2014 Intelligent transport systems - Lane keeping assistance systems (LKAS) - Performance requirements and test procedures*. 2014
- [ISO 15622 2010] INTERNATIONAL ORGANIZATION FOR STANDARDIZATION (ISO): *ISO 15622:2010 Intelligent transport systems - Adaptive Cruise Control systems - Performance requirements and test procedures*. 2010
- [ISO 22179 2009] INTERNATIONAL ORGANIZATION FOR STANDARDIZATION (ISO): *ISO 22179:2009 Intelligent transport systems - Full speed range adaptive cruise control (FSRA) systems - Performance requirements and test procedures*. 2009
- [ISO 26262 2011] INTERNATIONAL ORGANIZATION FOR STANDARDIZATION (ISO): *ISO 26262:2011 Road vehicles - Functional safety*. 2011
- [ISO 9000 2000] INTERNATIONAL ORGANIZATION FOR STANDARDIZATION (ISO): *ISO 9000:2000 Quality management systems - Fundamentals and vocabulary*. 2000
- [Junghanns u. a. 2008] JUNGHANNS, A.; MAUSS, J.; TATAR, M.: TestWeaver - Funktionstest nach dem Schachspielerprinzip. In: *Autotest*, 2008
- [Kacher 2016] KACHER, G.: In der neuen E-Klasse steckt Hightech, die kein Vertrauen schafft. In: *Süddeutsche Zeitung*, URL <http://www.sueddeutsche.de/auto/mittelklasselimousine-im-fahrbericht-in-der-neue-e-klasse-steckt-hightech-die-kein-vertrauen-schafft-1.2899893>. – Zugriffsdatum: 18.05.2016, 2016
- [Kacker u. a. 2013] KACKER, R. N.; KUHN, D. R.; LEI, Y.; LAWRENCE, J. F.: Combinatorial testing for software: An adaptation of design of experiments. In: *Measurement* 46 (2013), Nr. 9, S. 3745–3752
- [Kaiser 2015a] KAISER, B.: *Persönliche Kommunikation: Aufwand für evolutionäres Testen*. 2015
- [Kaiser 2015b] KAISER, B.: *Persönliche Kommunikation: Testabdeckung bei wertkontinuierlichen Parametern*. 2015
- [Kemper 2006] KEMPER, C.: *Dynamische Verkehrsumlegung mit statischen Matrizen in einem mikroskopischen Simulationsmodell*, Universität Hannover, Dissertation, 2006
- [Klein 2014] KLEIN, B.: *Versuchsplanung - DoE: Einführung in die Taguchi/Shainin-Methodik*. Bd. 4. München : De Gruyter Oldenbourg, 2014
- [Kohn 2005] KOHN, W.: *Statistik: Datenanalyse und Wahrscheinlichkeitsrechnung*. Berlin, Heidelberg : Springer Vieweg, 2005
- [Korosec 2016] KOROSEC, K.: *Google Preparing to Expand Self-Driving Car Program to Four More Cities*. 2016. – URL <http://fortune.com/2016/01/29/google-self-driving-cars-cities/>. – Zugriffsdatum: 12.05.2016



- [Krajzewicz u. a. 2012] KRAJZEWICZ, D.; ERDMANN, J.; BEHRISCH, M.; BIEKER, L.: Recent Development and Applications of SUMO - Simulation of Urban MObility. In: *International Journal On Advances in Systems and Measurements* 5 (2012), Nr. 3&4, S. 128–138
- [Kruse u. a. 2009] KRUSE, P. M.; WEGENER, J.; WAPPLER, S.: A Highly Configurable Test System for Evolutionary Black-box Testing of Embedded Systems. In: *11th Annual Conference on Genetic and Evolutionary Computation*, 2009, S. 1545–1552
- [Kuhn u. a. 2004] KUHN, D. R.; WALLACE, D. R.; GALLO JR., A. M.: Software Fault Interactions and Implications for Software Testing. In: *IEEE Transactions on Software Engineering* 30 (2004), Nr. 6, S. 418–421
- [Kuhn u. a. 2013] KUHN, D. R.; KACKER, R. N.; LEI, Y.: *Introduction to Combinatorial Testing*. CRC Press Taylor & Francis Group, 2013
- [Lackes u. a. 2016] LACKES, R.; SIEPERMANN, M.; ROTTMANN, H.; AUER, B. R.; LÜBBECKE, M.: *Gabler Wirtschaftslexikon, Stichwort: Simulation*. 2016. – URL <http://wirtschaftslexikon.gabler.de/Archiv/55029/simulation-v13.html>. – Zugriffsdatum: 12.02.2016
- [Lackmann 2015] LACKMANN, M.: *Entwicklung und Implementierung eines Frameworks zur Nachbildung prinzipbedingter Messfehler von Sensordaten in der Simulation*, Technische Universität Braunschweig, Masterthesis, 2015
- [Laschinsky u. a. 2010] LASCHINSKY, Y.; NEUMANN-COSEL, K. von; GONTER, M.; WEGWERTH, C.; DUBITZKY, R.; KNOLL, A.: Evaluation of an Active Safety Light using Virtual Test Drive within Vehicle in the Loop. In: *2010 IEEE International Conference on Industrial Technology (ICIT)*. Vi a del Mar, Chile : IEEE, 2010, S. 1112–1119
- [Law und Kelton 1999] LAW, A. M.; KELTON, D. M.: *Simulation Modeling and Analysis*. Bd. 3. Boston : McGraw-Hill Higher Education, 1999
- [Lei u. a. 2008] LEI, Y.; KACKER, R.; KUHN, D. R.; OKUN, V.; LAWRENCE, J.: IPOG/IPOG-D: efficient test generation for multi-way combinatorial testing. In: *Software Testing, Verification and Reliability* 18 (2008), Nr. 3, S. 125–148
- [Leveson 2011] LEVESON, N.: *Engineering a Safer World: Systems Thinking Applied to Safety*. MIT Press, 2011
- [Levinson u. a. 2007] LEVINSON, J.; MONTEMERLO, M.; THRUN, S.: Map-Based Precision Vehicle Localization in Urban Environments. In: *Proceedings of Robotics: Science and Systems*. Atlanta, GA, USA, 2007
- [Liggesmeyer 2009] LIGGESMEYER, P.: *Software-Qualität: Testen, Analysieren und Verifizieren von Software*. Bd. 2. Heidelberg : Spektrum Akademischer Verlag, 2009
- [Lighthill und Whitham 1955] LIGHTHILL, M. J.; WHITHAM, G. B.: On Kinematic Waves. II. A Theory of Traffic Flow on Long Crowded Roads. In: *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 229 (1955), Nr. 1178, S. 317–345

- [Lindgren u. a. 2008] LINDGREN, A. M.; CHEN, F.; JORDAN, P. W.; ZHANG, H.: Requirements for the Design of Advanced Driver Assistance Systems – The Differences between Swedish and Chinese Drivers. In: *International Journal of Design 2* (2008), Nr. 2, S. 41–54
- [Lindlar 2012] LINDLAR, F.: *Modellbasierter evolutionärer Funktionstest*, Technische Universität Berlin, Dissertation, 2012
- [Lippold u. a. 2014] LIPPOLD, C.; SCHWOON, G.; ROHLOFF, M.; BUSCH, F.; WEHNER-HEIL, A.; FLOSS, R.; HALFMANN, U.; STÖCKNER, M.; PORWOLLIK, J.; FRIEBEL, W.; COHRS, H.; HOLL, G.; OBERLÄNDER, A.; FÜHRLING, S.: *DER ELSNER 2014 - Handbuch für Straßen- und Verkehrswesen*. 2014
- [Malz 2013] MALZ, C.: Agentenbasierte dynamische Testfallpriorisierung. In: GÖHNER, P. (Hrsg.): *Agentensysteme in der Automatisierungstechnik*. Berlin Heidelberg : Springer Vieweg, 2013, S. 275–290
- [Marek 2014] MAREK, G.: *Entwicklung und Implementierung eines Frameworks zur systematischen Gestaltung und Variation von virtuellen Umgebungen in einer Fahrsimulation zur Testung von Fahrerassistenzsystemen*, Technische Universität Braunschweig, Masterthesis, 2014
- [Matthaei u. a. 2015] MATTHAEI, R.; RESCHKA, A.; RIEKEN, J.; DIERKES, F.; ULBRICH, S.; WINKLE, T.; MAURER, M.: Automones Fahren. In: WINNER, H. (Hrsg.); HAKULI, S. (Hrsg.); LOTZ, F. (Hrsg.); SINGER, C. (Hrsg.): *Handbuch Fahrerassistenzsysteme: Grundlagen, Komponenten und Systeme für aktive Sicherheit und Komfort* Bd. 3. Wiesbaden : Springer Vieweg, 2015, S. 1139–1166
- [Matthaei 2015] MATTHAEI, R.: *Wahrnehmungsgestützte Lokalisierung in fahrstreifengenauen Karten für Fahrerassistenzsysteme und automatisches Fahren in urbaner Umgebung*, Technische Universität Braunschweig, Dissertation, 2015
- [Maurer 2013] MAURER, M.: *Moderation auf der AAET 2013: Herausforderung des Testens zukünftiger automatisierter Fahrfunktionen*. 2013
- [Maurer 2015a] MAURER, M.: Einleitung. In: MAURER, M. (Hrsg.); GERDES, J. C. (Hrsg.); LENZ, B. (Hrsg.); WINNER, H. (Hrsg.): *Autonomes Fahren* Bd. 1. Heidelberg : Springer Vieweg, 2015, S. 1–8
- [Maurer 2015b] MAURER, M.: *Skript zur Vorlesung: Fahrerassistenzsysteme mit maschineller Wahrnehmung*. 2015
- [Maurer und Reschka 2015] MAURER, M.; RESCHKA, A.: *Persönliche Kommunikation: Anforderungen durch hochautomatisierte Fahrfunktionen*. 2015
- [Maurer und Winner 2013] MAURER, M.; WINNER, H. ; MAURER, M. (Hrsg.); WINNER, H. (Hrsg.): *Automotive Systems Engineering*. Bd. 1. Berlin, Heidelberg : Springer-Verlag Berlin Heidelberg, 2013

- [Mauritz u. a. 2015] MAURITZ, M.; HOWAR, F.; RAUSCH, A.: From Simulation to Operation: Using Design Time Artifacts to Ensure the Safety of Advanced Driving Assistance Systems at Runtime. In: *International Workshop on Modeling in Automotive Software Engineering 2015*. Ottawa, Kanada, 2015, S. 43–52
- [Meinecke 2016] MEINECKE, M.: *Persönliche Kommunikation: Ausbreitungseffekte von Radarsignalen*. 2016
- [Menzel 2016] MENZEL, T.: *Persönliche Kommunikation: Logische und konkrete Szenarien*. 2016
- [Meyers 2009] MEYERS, R. A.: *Encyclopedia of Complexity and Systems Science*. Bd. 1. New York : Springer-Verlag New York, 2009
- [Mirwaldt u. a. 2012] MIRWALDT, P.; BARTELS, A.; TO, T.; PASCHEKA, P.; LEMMER, K.; HUHLE, R.; MALBERG, H.; ZAUNSEDER, S.: Gestaltung eines Notfallassistentensystems bei medizinisch bedingter Fahruntfähigkeit. In: *5. Tagung Fahrerassistenz*. München, 2012
- [Morris und Roth 1982] MORRIS, M. F.; ROTH, P. F.: *Computer performance evaluation: tools and techniques for effective analysis*. Van Nostrand Reinhold Company, 1982
- [Mueller u. a. 2015] MUELLER, S.; HOSPACH, D.; BRINGMANN, O.; GERLACH, J.; ROSENSTIEL, W.: Robustness Evaluation and Improvement for Vision-Based Advanced Driver Assistance Systems. In: *18th International Conference on Intelligent Transportation Systems (ITSC)*, 2015, S. 2659–2664
- [Nagel und Schreckenberg 1992] NAGEL, K.; SCHRECKENBERG, M.: A cellular automaton model for freeway traffic. In: *Journal de Physique I* 2 (1992), Nr. 12, S. 2221–2229
- [Nelson u. a. 2005] NELSON, P.; WLUDYKA, P.; COPELAND, K.: *The Analysis of Means: A Graphical Method for Comparing Means, Rates, and Proportions*. Bd. 1. Philadelphia, PA, USA : Society for Industrial and Applied Mathematics, 2005
- [Nentwig 2013] NENTWIG, M.: *Untersuchungen zur Anwendung von computergenerierten Kamerabildern für die Entwicklung und den Test von Fahrerassistenzsystemen*, Friedrich-Alexander-Universität Erlangen-Nürnberg, Dissertation, 2013
- [Nentwig und Stamminger 2010] NENTWIG, M.; STAMMINGER, M.: A method for the reproduction of vehicle test drives for the simulation based evaluation of image processing algorithms. In: *13th International IEEE Conference on Intelligent Transportation Systems (ITSC)*. Funchal, Portugal : IEEE, 2010, S. 1307–1312
- [Nentwig und Stamminger 2011] NENTWIG, M.; STAMMINGER, M.: Hardware-in-the-loop testing of computer vision based driver assistance systems. In: *2011 IEEE Intelligent Vehicles Symposium (IV)*. Baden-Baden, Deutschland : IEEE, 2011, S. 339–344
- [von Neumann-Cosel 2013] NEUMANN-COSEL, K. von: *Virtual Test Drive - Simulation umfeldbasierter Fahrzeugfunktionen*, Technische Universität München, Dissertation, 2013

- [Nie und Leung 2011] NIE, C.; LEUNG, H.: A survey of combinatorial testing. In: *ACM Computing Surveys* 43 (2011), Nr. 2, S. 1–29
- [NRC 2010] NATIONAL RESEARCH COUNCIL; BOARD, T. R.: *Highway Capacity Manual (HCM) 2010*. 2010
- [Petit 2015] PETIT, J.: *Präsentation: Self-Driving and Connected Cars: Fooling Sensors and Tracking Drivers*. 2015
- [Pfeffer u. a. 2015] PFEFFER, R. J.; LEICHSENRING, T.; SCHWAB, S.: Vehicle-in-the-Loop as a Method to Tangibly Experience Active Safety Systems at an Early Stage. In: *International Symposium on Future Active Safety Technology - Toward zero traffic accidents*. Göteborg, Schweden, 2015, S. 515–524
- [Plättner u. a. 2016] PLÄTTNER, J.; RICHTER, A.; KÖSTER, F.; BARTELS, A.; MOCK, H.; SCHÖNER, H.; SCHIEMENTZ, M.; EBERLE, U.; STEININGER, U.; STILLER, M.: *Vorhabenbeschreibung PEGASUS - Projekt zur Etablierung von generell akzeptierten Gütekriterien, Werkzeugen und Methoden sowie Szenarien und Situationen zur Freigabe hochautomatisierter Fahrfunktionen*. 2016
- [Prialé Olivares u. a. 2016] PRIALÉ OLIVARES, S.; REBERNIK, N.; EICHBERGER, A.; STADLOBER, E.: *Virtual Stochastic Testing of Advanced Driver Assistance Systems*. S. 25–35. In: SCHULZE, T. (Hrsg.); MÜLLER, B. (Hrsg.); MEYER, G. (Hrsg.): *Advanced Microsystems for Automotive Applications 2015*. Cham, Schweiz : Springer International Publishing, 2016
- [PTV Group ] PTV GROUP: *PTV Vissim*. – URL <http://vision-traffic.ptvgroup.com/de/produkte/ptv-vissim/>. – Zugriffsdatum: 20.10.2015
- [Reschka u. a. 2015a] RESCHKA, A.; BAGSCHIK, G.; ULBRICH, S.; NOLTE, M.; MAURER, M.: The ability and skill graphs for system modeling, online monitoring, and decision support for vehicle guidance systems. In: *2015 IEEE Intelligent Vehicles Symposium (IV)*. Seoul, Südkorea : IEEE, 2015, S. 933–939
- [Reschka u. a. 2012] RESCHKA, A.; BÖHMER, J. R.; NOTHDURFT, T.; HECKER, P.; LICHTHE, B.; MAURER, M.: A surveillance and safety system based on performance criteria and functional degradation for an autonomous vehicle. In: *15th International IEEE Conference on Intelligent Transportation Systems (ITSC)*. Anchorage, AK, USA : IEEE, 2012, S. 237–242
- [Reschka 2015] RESCHKA, A.: *Persönliche Kommunikation: Kombination von ACC und LKAS als teilautomatisierte Fahrfunktion*. 2015
- [Reschka 2016a] RESCHKA, A.: *Fertigkeiten- und Fähigkeitengraphen als Grundlage des sicheren Betriebs von automatisierten Fahrzeugen im öffentlichen Straßenverkehr in städtischer Umgebung*, Technische Universität Braunschweig, Dissertation, 2016. – angekündigt
- [Reschka 2016b] RESCHKA, A.: *Persönliche Kommunikation: Bedeutung der Funktionspezifikation bei der Auswahl von Einflussparametern*. 2016

- [Reschka und Maurer 2015] RESCHKA, A.; MAURER, M.: Conditions for a safe state of automated road vehicles. In: *it - information technology* 57 (2015), Nr. 4, S. 215–222
- [Reschka u. a. 2015b] RESCHKA, A.; RIEKEN, J.; MAURER, M.: Entwicklungsprozess von Kollisionsschutzsystemen für Frontkollisionen: Systeme zur Warnung, zur Unfallschwereminderung und zur Verhinderung. In: WINNER, H. (Hrsg.); HAKULI, S. (Hrsg.); LOTZ, F. (Hrsg.); SINGER, C. (Hrsg.): *Handbuch Fahrerassistenzsysteme* Bd. 3. Wiesbaden : Springer Vieweg, 2015, S. 913–935
- [Rieken u. a. 2015] RIEKEN, J.; MATTHAEI, R.; MAURER, M.: Toward Perception-Driven Urban Environment Modeling for Automated Road Vehicles. In: *18th International Conference on Intelligent Transportation Systems (ITSC)*. Las Palmas, Spanien : IEEE, 2015, S. 731–738
- [Rieken und Maurer 2016] RIEKEN, J.; MAURER, M.: Sensor scan timing compensation in environment models for automated road vehicles. In: *19th International Conference on Intelligent Transportation Systems (ITSC)*. Rio de Janeiro, Brasilien : IEEE, 2016, S. 635–642
- [Robinson-Mallet u. a. 2010] ROBINSON-MALLET, C.; WEGENER, J.; HEERS, H.; LIGGESMEYER, P.: Integration und Validation von Produktlinien für Fahrerassistenzsysteme im Kontext der ISO 26262. In: *26. VDI/VW-Gemeinschaftstagung - Fahrerassistenz und Integrierte Sicherheit* Bd. 26. Wolfsburg : VDI Verlag GmbH, 2010, S. 407–423
- [Rohde & Schwarz 2015] ROHDE & SCHWARZ: *Simplify pulse and emitter generation for radar testing*. 2015. – URL [http://cdn.rohde-schwarz.com/pws/dl\\_downloads/dl\\_application/pdfs/SMW200A\\_Radar-Testing\\_ac\\_3607-1742\\_92\\_v0100.pdf](http://cdn.rohde-schwarz.com/pws/dl_downloads/dl_application/pdfs/SMW200A_Radar-Testing_ac_3607-1742_92_v0100.pdf). – Zugriffsdatum: 12.05.2016
- [Rook 1986] ROOK, P.: Controlling software projects. In: *Software Engineering Journal* 1 (1986), Nr. 1, S. 7–16
- [Rose 2003] ROSE, M.: *Modellbildung und Simulation von Autobahnverkehr*, Universität Hannover, Dissertation, 2003
- [Roth 2015] ROTH, E.: *Persönliche Kommunikation: Virtuelle Sensormodelle*. 2015
- [Rupprecht 2014] RUPPRECHT, W.: *Einführung in die Theorie der kognitiven Kommunikation*. 1. Wiesbaden : Springer Fachmedien Wiesbaden, 2014
- [SAE 2014] SOCIETY OF AUTOMOTIVE ENGINEERS (SAE): Taxonomy and Definitions for Terms Related to On-Road Motor Vehicle Automated Driving Systems. In: *Surface Vehicle Information Report* (2014), Nr. J3016
- [SAE 2015] SOCIETY OF AUTOMOTIVE ENGINEERS (SAE): Guidelines for Safe On-Road Testing of SAE Level 3, 4, and 5 Prototype Automated Driving Systems (ADS). In: *Surface Vehicle Information Report* (2015), Nr. J3018



- [Saust u. a. 2011] SAUST, F.; WILLE, J. M.; LICHTTE, B.; MAURER, M.: Autonomous Vehicle Guidance on Braunschweig's inner ring road within the Stadtpilot Project. In: *2011 IEEE Intelligent Vehicles Symposium (IV)*. Baden-Baden, Deutschland : IEEE, 2011, S. 169–174
- [Saust u. a. 2009] SAUST, F.; MÜLLER, T.; WILLE, J. M.; MAURER, M.: Entwicklungsbegleitendes Simulations- und Testkonzept für autonome Fahrzeuge in städtischen Umgebungen. In: *AAET - Automatisierungssysteme, Assistenzsysteme und eingebettete Systeme für Transportmittel*. Braunschweig, Deutschland : ITS Niedersachsen, 2009, S. 87–107
- [Schaub 1996] SCHAUB, H.: Exception error: Über Fehler und deren Ursachen beim Handeln in Unbestimmtheit und Komplexität. In: *Gdi-Impuls* 14 (1996), Nr. 4, S. 3–16
- [Schmidt 2012] SCHMIDT, F.: *Funktionale Absicherung kamerabasierter Aktiver Fahrerassistenzsysteme durch Hardware-in-the-Loop-Tests*, Universität Kaiserslautern, Dissertation, 2012
- [Scholz u. a. 2014] SCHOLZ, S.; CHLOSTA, S.; FRETER, S.; SCHULDT, F.: Prototypische Umsetzung eines Bau- und Engstellenassistenten. In: *AAET - Automatisierungssysteme, Assistenzsysteme und eingebettete Systeme für Transportmittel*. Braunschweig, Deutschland : ITS Niedersachsen, 2014, S. 291–314
- [Schöner u. a. 2011] SCHÖNER, H.; HURICH, W.; LUTHER, J.; HERRTWICH, R. G.: Koordiniertes automatisiertes Fahren für die Erprobung von Assistenzsystemen. In: *ATZ - Automobiltechnische Zeitschrift* 113 (2011), Nr. 1, S. 40–45
- [Schreiber 2012] SCHREIBER, M.: *Konzeptionierung und Evaluierung eines Ansatzes zu einer manöverbasierten Fahrzeugführung im Nutzungskontext Autobahnfahrten*, Technische Universität Darmstadt, Dissertation, 2012
- [Schuldt 2014] SCHULDT, F.: *Workshop Fahrzeugsystemtechnik: Dynamische Objekte im modularen virtuellen Testbaukasten*. 2014
- [Schuldt u. a. 2014] SCHULDT, F.; LICHTTE, B.; MAURER, M.; SCHOLZ, S.: Systematische Auswertung von Testfällen für Fahrfunktionen im modularen virtuellen Testbaukasten. In: *9. Workshop Fahrerassistenzsysteme*. Walting, Deutschland : Uni-DAS e.v., 2014, S. 169–179
- [Schuldt u. a. 2015a] SCHULDT, F.; MENZEL, T.; DUPUIS, M.; KRAUSE, J.; SCHMIDT, B.; SYMKENBERG, K.: *Workshop OpenScenario*. 2015
- [Schuldt u. a. 2015b] SCHULDT, F.; MENZEL, T.; MAURER, M.: Eine Methode für die Zuordnung von Testfällen für automatisierte Fahrfunktionen auf X-in-the-Loop Verfahren im modularen virtuellen Testbaukasten. In: *10. Workshop Fahrerassistenzsysteme*. Walting, Deutschland : Uni-DAS e.v., 2015, S. 171–182



- [Schuldt u. a. 2016] SCHULDT, F.; RESCHKA, A.; MAURER, M.: A Method for an Efficient Systematic Test Case Generation for Advanced Driver Assistance Systems in Virtual Environments. In: MAURER, M. (Hrsg.); WINNER, H. (Hrsg.); PROKOP, G. (Hrsg.): *Automotive Systems Engineering* Bd. 2. Springer-Verlag Berlin Heidelberg, 2016. – angenommen zur Veröffentlichung
- [Schuldt u. a. 2013] SCHULDT, F.; SAUST, F.; LICHTER, B.; MAURER, M.; SCHOLZ, S.: Effiziente systematische Testgenerierung für Fahrerassistenzsysteme in virtuellen Umgebungen. In: *AAET - Automatisierungssysteme, Assistenzsysteme und eingebettete Systeme für Transportmittel*. Braunschweig, Deutschland : ITS Niedersachsen, 2013, S. 114 – 134
- [Schulze 2014] SCHULZE, H.: *Entwicklung und Bewertung einer kombinatorischen Testfallgenerierung für Fahrerassistenzsysteme*, Otto-von-Guericke-Universität Magdeburg, Bachelorarbeit, 2014
- [Schweigert 2003] SCHWEIGERT, M.: *Fahrerblickverhalten und Nebenaufgaben*, Technische Universität München, Dissertation, 2003
- [Shimada u. a. 2015] SHIMADA, H.; YAMAGUCHI, A.; TAKADA, H.; SATO, K.: Implementation and Evaluation of Local Dynamic Map in Safety Driving Systems. In: *Journal of Transportation Technologies* 5 (2015), Nr. 2, S. 102–112
- [Siebertz u. a. 2010] SIEBERTZ, K.; BEBBER, D. van; HOCHKIRCHEN, T.: *Statistische Versuchsplanung*. Bd. 1. Berlin Heidelberg : Springer Berlin Heidelberg, 2010
- [Sommerville 2006] SOMMERVILLE, I.: *Software Engineering*. 8. Boston, MA, USA : Addison-Wesley Longman Publishing Co., Inc., 2006
- [Spillner und Linz 2012] SPILLNER, A.; LINZ, T.: *Basiswissen Softwaretest - Aus- und Weiterbildung zum Certified Tester, Foundation Level nach ISTQB-Standard*. 3. Heidelberg : dpunkt.verlag GmbH, 2012
- [Statistisches Bundesamt 2016] STATISTISCHES BUNDESAMT: Verkehrsunfälle Zeitreihen 2014. Wiesbaden, 2016. – Forschungsbericht
- [Stiller 2005] STILLER, C.: Fahrerassistenzsysteme - Von realisierten Funktionen zum vernetzt wahrnehmenden, selbstorganisierenden Verkehr. In: MAURER, M. (Hrsg.); STILLER, C. (Hrsg.): *Fahrerassistenzsysteme mit maschineller Wahrnehmung* Bd. 1. Berlin Heidelberg : Springer-Verlag, 2005, S. 1–20
- [Strasser 2012] STRASSER, B.: *Vernetzung von Test- und Simulationenmethoden für die Entwicklung von Fahrerassistenzsystemen*, Technische Universität München, Dissertation, 2012
- [Symkenberg 2014] SYMKENBERG, K.: *Entwicklung und Implementierung eines Frameworks zur systematischen Gestaltung und Variation von dynamischen Szenarien zur Testung von Fahrerassistenzsystemen*, Universität Hannover, Masterthesis, 2014

- [Taguchi u. a. 2005] TAGUCHI, G.; CHOWDHURY, S.; WU, Y.; TAGUCHI, S.; YANO, H.: *Taguchi's quality engineering handbook*. Kanada : John Wiley & Sons, Inc., Hoboken, New Jersey, 2005
- [Tatar und Mauss 2014] TATAR, M.; MAUSS, J.: Systematic Test and Validation of Complex Embedded Systems. In: *Embedded Real Time Software and Systems (ERTS)*. Toulouse, Frankreich, 2014, S. 1–10
- [Thomason und Gonzalez 1985] THOMASON, M. G.; GONZALEZ, R. C.: Data Structures and Databases in Digital Scene Analysis. In: TOU, J. T. (Hrsg.): *Advances in Information Systems Science* Bd. 9. New York, N.Y., USA : Springer US, 1985, S. 1–47
- [Thommen u. a. 2015] THOMMEN, J.; AMELUNG, o. E.; MÜHLBACHER, A.; KRAUTH, C.: *Gabler Wirtschaftslexikon, Stichwort: Effektivität*. 2015. – URL <http://wirtschaftslexikon.gabler.de/Archiv/7639/effektivitaet-v9.html>. – Zugriffsdatum: 21.11.2015
- [Tideman 2010] TIDEMAN, M.: Scenario-based simulation environment for assistance systems. In: *AutoTechnology* 10 (2010), Nr. 1, S. 28–32
- [Tölle 1996] TÖLLE, W.: *Ein Fahrmanöverkonzept für einen maschinellen Kopiloten*. Düsseldorf : VDI-Verlag, 1996
- [Treiber u. a. 2000] TREIBER, M.; HENNECKE, A.; HELBING, D.: Congested Traffic States in Empirical Observations and Microscopic Simulations. In: *Physical Review E* 62 (2000), Nr. 2, S. 1805–1824
- [TRM 2013] DEPARTMENT OF TRANSPORT AND MAIN ROADS: *Road planning and design manual*. 2013
- [Ulbrich 2016] ULBRICH, S.: *Towards Tactical Lane Change Behavior Planning for Automated Vehicles*, Technische Universität Braunschweig, Dissertation, 2016. – angekündigt
- [Ulbrich u. a. 2015] ULBRICH, S.; MENZEL, T.; RESCHKA, A.; SCHULDT, F.; MAURER, M.: Definition der Begriffe Szene , Situation und Szenario für das automatisierte Fahren. In: *10. Workshop Fahrerassistenzsysteme*. Walting, Deutschland : Uni-DAS e.v., 2015, S. 105–117
- [Ulbrich u. a. 2016] ULBRICH, S.; SCHULDT, F.; MENZEL, T.; HOMEIER, K.; STEINHOFF, M.; KRAUSE, J.; MAURER, M.: Testing and Validating Tactical Lane Change Behavior Planning for Automated Driving. In: HORN, M. (Hrsg.); WATZENIG, D. (Hrsg.): *Automated Driving*. Springer International Publishing AG, 2016, S. 451–471
- [VDA 2015] VERBAND DER AUTOMOBILINDUSTRIE E.V. (VDA): *Automatisierung - Von Fahrerassistenzsystemen zum automatisierten Fahren*. 2015
- [VDI 2206 2004] VEREIN DEUTSCHER INGENIEURE (VDI): *VDI-Richtlinien VDI 2206: Entwicklungsmethodik für mechatronische Systeme*. 2004
- [VDI 3633 2013] VEREIN DEUTSCHER INGENIEURE (VDI): *VDI-Richtlinie VDI3633: Simulation von Logistik-, Materialfluss- und Produktionssystemen - Grundlagen*. 2013

- [Verhoeff u. a. 2000] VERHOEFF, L.; VERBURG, D. J.; LUPKER, H. A.; KUSTERS, L. J.: VEHIL: A full-scale test methodology for intelligent transport systems, vehicles and subsystems. In: *IEEE Intelligent Vehicles Symposium (IV)*. Dearborn, MI, USA : IEEE, 2000, S. 369–375
- [Volkswagen AG 2011] VOLKSWAGEN AG: *Datenblatt: Passat CC*. 2011. – URL [http://www.volkswagen.de/content/medialib/vwd4/de/dialog/pdf/passat\\_cc/passat\\_cc\\_tup\\_20110113/\\_jcr\\_content/renditions/rendition.file/passat\\_cc\\_tup\\_20110113.pdf](http://www.volkswagen.de/content/medialib/vwd4/de/dialog/pdf/passat_cc/passat_cc_tup_20110113/_jcr_content/renditions/rendition.file/passat_cc_tup_20110113.pdf)
- [Wachenfeld und Winner 2015a] WACHENFELD, W.; WINNER, H.: Die Freigabe des autonomen Fahrens. In: MAURER, M. (Hrsg.); GERDES, J. C. (Hrsg.); LENZ, B. (Hrsg.); WINNER, H. (Hrsg.): *Autonomes Fahren*. Heidelberg : Springer Vieweg, 2015, S. 439–464
- [Wachenfeld und Winner 2015b] WACHENFELD, W.; WINNER, H.: Virtual Assessment of Automation in Field Operation - A New Runtime Validation Method. In: *10. Workshop Fahrerassistenzsysteme*. Walting, Deutschland : Uni-DAS e.v., 2015, S. 161–170
- [Wachenfeld u. a. 2015] WACHENFELD, W.; WINNER, H.; GERDES, J. C.; LENZ, B.; MAURER, M.; BEIKER, S.; FRAEDRICH, E.; WINKLE, T.: Use-Cases des autonomen Fahrens. In: MAURER, M. (Hrsg.); GERDES, J. C. (Hrsg.); LENZ, B. (Hrsg.); WINNER, H. (Hrsg.): *Autonomes Fahren*. Heidelberg : Springer Vieweg, 2015, S. 9–37
- [Wegener 2003] WEGENER, I.: *Komplexitätstheorie*. 1. Berlin, Heidelberg : Springer-Verlag, 2003
- [Weilkes 2000] WEILKES, M.: *Auslegung und Analyse von Fahrerassistenzsystemen mittels Simulation*, RWTH Aachen, Dissertation, 2000
- [Wiedemann 1974] WIEDEMANN, R.: *Simulation des Straßenverkehrsflusses*. Karlsruhe, Dissertation, 1974
- [Winner und Schopper 2015] WINNER, H.; SCHOPPER, M.: Adaptive Cruise Control. In: WINNER, H. (Hrsg.); HAKULI, S. (Hrsg.); LOTZ, F. (Hrsg.); SINGER, C. (Hrsg.): *Handbuch Fahrerassistenzsysteme: Grundlagen, Komponenten und Systeme für aktive Sicherheit und Komfort* Bd. 3. Wiesbaden : Springer Vieweg, 2015, S. 851–891
- [Winner 2002] WINNER, H.: *Einrichtung zum Bereitstellen von Signalen in einem Kraftfahrzeug*. WO Patent App. PCT/DE2001/004,569. 2002
- [Winner 2003] WINNER, H.: Die lange Entwicklung von ACC. In: STILLER, C. (Hrsg.); MAURER, M. (Hrsg.): *Workshop Fahrerassistenzsysteme FAS2003*. Karlsruhe : Freundeskreis Mess- und Regelungstechnik Karlsruhe e.V., 2003, S. 1–5
- [Winner 2015a] WINNER, H. ; WINNER, H. (Hrsg.); HAKULI, S. (Hrsg.); LOTZ, F. (Hrsg.); SINGER, C. (Hrsg.): *Handbuch Fahrerassistenzsysteme*. 3. Wiesbaden : Springer Vieweg, 2015

- [Winner 2015b] WINNER, H.: Radarsensorik. In: WINNER, H. (Hrsg.); HAKULI, S. (Hrsg.); LOTZ, F. (Hrsg.); SINGER, C. (Hrsg.): *Handbuch Fahrerassistenzsysteme* Bd. 3. Wiesbaden : Springer Vieweg, 2015, S. 259–316
- [Winner 2016] WINNER, H.: *Persönliche Kommunikation: Voraussetzungen für eine Metrik*. 2016
- [Wurster u. a. 2010] WURSTER, U.; ORTLECHNER, M.; SCHICK, B.; DRENTHE, E.; CRAWLEY, J.: Simulationsbasierte Homologation von Bremsen für Lkw-Anhänger. In: *ATZ - Automobiltechnische Zeitschrift* 112 (2010), S. 2–7
- [Yuen u. a. 2008] YUEN, A.; BROWN, C.; ZOTT, C.; HILLER, A.; AHLERS, F.; WEVERS, K.; DREHER, S.; SCHENDZIELORZ, T.; BARTELS, C.; PAPP, Z.; NETTEN, B.: Local dynamic map specification. 2008. – Forschungsbericht
- [Zeeb 2015] ZEEB, E.: Daimler's New Full-Scale, High-dynamic Driving Simulator – A Technical Overview . In: *Driving Simulation Conference*. Tübingen, Deutschland, 2015, S. 157–165
- [Ziegenbalg u. a. 2016] ZIEGENBALG, J.; ZIEGENBALG, O.; ZIEGENBALG, B.: *Algorithmen von Hammurapi bis Gödel*. Wiesbaden : Springer Spektrum, 2016
- [Ziegler u. a. 2014] ZIEGLER, J.; BENDER, P.; SCHREIBER, M.; LATEGAHN, H.; STRAUSS, T.; STILLER, C.; DANG, T.; FRANKE, U.; APPENRODT, N.; KELLER, C. G.; KAUS, E.; HERRTWICH, R. G.; RABE, C.; PFEIFFER, D.; LINDNER, F.; STEIN, F.; ERBS, F.; ENZWEILER, M.; KNOPPEL, C.; HIPPE, J.; HAUEIS, M.; TREPTE, M.; BRENN, C.; TAMKE, A.; GHANAAT, M.; BRAUN, M.; JOOS, A.; FRITZ, H.; MOCK, H.; HEIN, M.; ZEEB, E.: Making Bertha Drive - An Autonomous Journey on a Historic Route. In: *IEEE Intelligent Transportation Systems Magazine* 6 (2014), Nr. 2, S. 8–20
- [Zofka u. a. 2015] ZOFKA, M. R.; KUHNT, F.; KOHLHAAS, R.; RIST, C.; SCHAMM, T.; MARIUS, J. Z.: Data-Driven Simulation and Parametrization of Traffic Scenarios for the Development of Advanced Driver Assistance Systems. In: *International Conference on Information Fusion*. Washington, DC, USA : IEEE, 2015, S. 1422–1428
- [Zulauf 2011] ZULAUF, C.: *Persönliche Kommunikation: Planung von Arbeitsstellen, Niedersächsische Landesbehörde für Straßenbau und Verkehr*. 2011



# Anhang





## A Detaillierte Ergebnisse der Fallstudie des Engstellenassistenten

Testfall	Verswenkungsmaß	Behelfsfahrfreifensbreite in m	Abstand der Randbebauung in m	Verswenkungsablage in m	Art der Randbebauung	Bestanden	Minimaler Abstand zur Randbebauung in m	Qualitätsverlust für den minimalen Abstand zur Randbebauung	Mittlerer minimaler Abstand zur Randbebauung in m	Qualitätsverlust für den mittleren minimalen Abstand zur Randbebauung	Maximale Ablage in m	Qualitätsverlust für die maximale Ablage	Mittlere Ablage in m	Qualitätsverlust für den mittleren Ablage	Maximaler Qualitätsverlust vom minimalen Abstand zur Randbebauung und maximalen Offset
1	1:5	2,5	5	3,5	Bake	✗	0	100	1,153	0,067	1,427	100	0,152	18,9	100
2	1:5	2,75	7	7	Kegel	✗	0	100	1,417	0,044	0,986	100	-0,017	0,1	100
3	1:5	3,00	9	10,5	Wand	✗	0	100	0,831	0,130	1,495	100	0,0631	1,1	100
4	1:5	3,25	11	14	Bake	✗	0	100	1,398	0,046	1,259	100	-0,092	1,6	100
5	1:5	3,50	13	3,5	Kegel	✗	0,229	1,7	2,523	0,014	1,430	100	-0,909	100	100
6	1:5	3,75	5	7	Wand	✗	0	100	1,428	0,044	1,857	100	0,205	4,4	100
7	1:7	2,50	7	10,5	Bake	✓	0,036	66,5	0,461	0,423	0,320	83,7	-0,063	3,3	83,7
8	1:7	2,75	9	14	Kegel	✓	0,053	32,0	0,475	0,397	0,304	41,0	0,068	2,1	41,0
9	1:7	3,00	11	3,5	Wand	✗	0	100	0,601	0,248	1,601	100	-0,007	0,0	100
10	1:7	3,25	13	7	Bake	✗	0	100	1,800	0,027	1,209	100	-0,381	27,7	100
11	1:7	3,50	5	10,5	Kegel	✗	0	100	1,007	0,088	1,980	100	0,440	26,9	100
12	1:7	3,75	7	14	Wand	✓	0,258	1,3	0,660	0,206	0,444	20,7	0,145	2,2	20,7
13	1:10	2,50	9	7	Bake	✓	0,186	2,5	0,444	0,454	0,323	85,6	-0,007	0,0	85,6
14	1:10	2,75	11	10,5	Kegel	✓	0,223	1,8	0,523	0,329	0,285	36,2	0,020	0,2	36,2
15	1:10	3,00	13	14	Wand	✓	0,157	3,6	0,416	0,518	0,257	18,5	0,170	8,1	18,5
16	1:10	3,25	5	3,5	Kegel	✗	0	100	0,751	0,159	1,883	100	1,372	100	100
17	1:10	3,50	7	3,5	Bake	✓	0,218	1,8	0,894	0,112	0,647	58,0	-0,098	1,4	58,0
18	1:10	3,75	9	3,5	Bake	✗	0	100	1,431	0,043	1,530	100	-0,801	67,6	100
19	1:15	2,50	11	7	Wand	✓	0,121	6,0	0,339	0,782	0,162	21,6	0,002	0,0	21,6

Testfall	Verschwenkungsmaß	Behelfsfahrstreifenbreite in m	Abstand der Randbebauung in m	Verschwenkungsablage in m	Art der Randbebauung	Bestanden	Minimaler Abstand zur Randbebauung in m	Qualitätsverlust für den minimalen Abstand zur Randbebauung	Mittlerer minimaler Abstand zur Randbebauung in m	Qualitätsverlust für den mittleren minimalen Abstand zur Randbebauung	Maximale Ablage in m	Qualitätsverlust für die maximale Ablage	Mittlere Ablage in m	Qualitätsverlust für den mittleren Ablage	Maximaler Qualitätsverlust vom minimalen Abstand zur Randbebauung und maximalen Offset
20	1:15	2,75	13	10,5	Bake	✗	0	100	0,866	0,119	1,110	100	-0,329	48,2	100
21	1:15	3,00	5	14	Kegel	✗	0,213	1,9	0,550	0,297	1,231	100	-0,620	100	100
22	1:15	3,25	7	10,5	Wand	✓	0,194	2,3	0,470	0,405	0,255	12,4	0,029	0,1	12,4
23	1:15	3,50	9	7	Wand	✓	0,215	1,9	0,508	0,347	0,309	13,3	0,047	0,3	13,3
24	1:15	3,75	11	3,5	Kegel	✗	0	100	0,740	0,164	2,109	100	0,090	0,8	100
25	1:20	2,50	13	14	Kegel	✓	0,096	9,5	0,414	0,524	0,250	51,2	-0,011	0,1	51,2
26	1:20	2,75	5	3,5	Wand	✓	0,032	87,1	0,369	0,659	0,159	11,3	0,002	0,0	87,1
27	1:20	3,00	7	7	Bake	✓	0,262	1,3	0,611	0,240	0,266	19,7	0,001	0,0	19,7
28	1:20	3,25	9	10,5	Wand	✓	0,212	2,0	0,476	0,397	0,240	11,0	0,012	0,0	11,0
29	1:20	3,50	11	14	Bake	✗	0	100	0,898	0,111	1,989	100	0,386	20,6	100
30	1:20	3,75	13	10,5	Kegel	✗	0	100	0,814	0,135	1,598	100	-0,177	3,3	100
31	1:25	2,50	5	7	Bake	✓	0,155	3,7	0,497	0,363	0,281	64,6	-0,082	5,6	64,6
32	1:25	2,75	7	14	Kegel	✓	0,157	3,6	0,482	0,386	0,228	23,2	0,030	0,4	23,2
33	1:25	3,00	9	3,5	Wand	✓	0,208	2,0	0,418	0,513	0,180	9,0	0,061	1,0	9,0
34	1:25	3,25	11	10,5	Kegel	✓	0,258	1,3	0,604	0,245	0,372	26,4	0,039	0,3	26,4
35	1:25	3,50	13	10,5	Kegel	✗	0	100	0,734	0,166	1,924	100	0,040	0,2	100
36	1:25	3,75	13	14	Bake	✗	0	100	0,710	0,178	0,968	98,6	-0,103	1,1	100

Tabelle A.1: Ergebnisse der Fallstudie des Engstellenassistenten